



TUGAS AKHIR - TE 145561

**PERANCANGAN KVARH METER UNTUK METERING
BEBAN INDUKTIF PADA JARINGAN TEGANGAN
MENENGAH**

Rahadian Yunanto Dwi Putra
NRP 2214038007

Dosen Pembimbing
Ir. Sjamsjul Anam, MT.
Fajar Budiman, ST., M.Sc.

PROGRAM STUDI TEKNIK LISTRIK
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017

-----Halaman ini sengaja dikosongkan-----



FINAL PROJECT - TE 145561

KVARH METER DESIGNING FOR INDUCTIVE LOAD METERING IN MEDIUM NETWORK

Rahadian Yunanto Dwi Putra
NRP 2214038007

Advisor
Ir. Sjamsjul Anam, MT.
Fajar Budiman, ST., M.Sc.

ELECTRICAL ENGINEERING STUDY PROGRAM
Electrical and Automation Engineering Department
Vocational Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2017

-----Halaman ini sengaja dikosongkan-----

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul **“Perancangan KVARH Meter Untuk Metering Beban Induktif Pada Jaringan Tegangan Menengah”**

adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 20 Juli 2017



Rahadian Yunanto Dwi
Putra
NRP 2214038007

-----Halaman ini sengaja dikosongkan-----

**PERANCANGAN KVARH METER UNTUK METERING BEBAN
INDUKTIF PADA JARINGAN TEGANGAN MENENGAH**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya
Pada**

**Program Studi Teknik Listrik
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing I

Dosen Pembimbing II

Ir. Sjamsul Anam, MT Fajar Budiman, ST., M.Sc.
NIP. 19630725 199003 1 002 NIP. 19860707 201404 1 001

**SURABAYA
JULI, 2017**

-----Halaman ini sengaja dikosongkan-----

PERANCANGAN KVARH METER UNTUK METERING BEBAN INDUKTIF PADA JARINGAN TEGANGAN MENENGAH

Nama : Rahadian Yunanto Dwi Putra
NRP : 2214038007
Pembimbing : Ir. Sjamsjul Anam, MT.
Fajar Budiman, ST., M.Sc.

ABSTRAK

Lebih dari 100.000 industri skala besar ada di Indonesia. Di ASEAN perindustrian Indonesia nomor 2 terbesar setelah Thailand. Dengan jumlah industri yang besar, maka jumlah pelanggan PLN semakin bertambah. Pelanggan industri adalah penyuplai pendapatan terbesar oleh PLN. Salah satu hal yang perlu diperhatikan pada pelanggan industri adalah jumlah pemakaian beban. Beban yang biasa digunakan oleh pelanggan industri adalah beban resistif dan induktif. Pada pemakaian beban induktif, pelanggan harus memperhatikan sudut cosphinya, sudut yang semakin kecil bisa dipastikan diakibatkan oleh kualitas beban induktif yang kurang baik. Batas nominal cosphi yang telah ditentukan PLN adalah 0,85. Jika pelanggan menggunakan beban induktif kurang dari 0,85 maka akan dikenakan denda oleh PLN. Pada Tugas Akhir ini alat yang dibuat dapat mengukur nominal VAR meter. Selain itu jika melihat permasalahan di lapangan tentang rawannya pelanggan terkena denda tagihan oleh PLN akibat pemakaian beban induktif. Dilengkapi pula media notifikasi dan rekapitulasi. Dengan begitu pemakaian beban induktif tersebut dapat dipantau dan diawasi oleh pelanggan. Penggunaan media notifikasi berupa internet yang terintegrasi dengan email pengguna dan nomor *handphone* pelanggan. Dengan pengiriman sampling gelombang yang dibaca oleh sensor arus dan tegangan dengan melalui sejumlah sampel dan membagi dengan jumlah sampel tersebut. Arduino mengolah gelombang dan mensampling gelombang *leading* dan *lagging* pada gelombang dan mengukur perbedaan cosphi sudutnya. Pada proses proses pengambilan data, ditemui pengukuran *error* untuk arus dan tegangan kurang dari 3%, *error* sudut cosphi error kurang dari 4%, dan *error* daya imajiner kurang dari 5%.

Kata Kunci : VAR, Induktif, Resistif, Daya, cosphi

-----Halaman ini sengaja dikosongkan-----

KVARH METER DESIGNING FOR INDUCTIVE LOAD METERING IN MEDIUM NETWORK

Name : Rahadian Yunanto Dwi Putra
NRP : 2214038007
Advisor : Ir. Sjamsjul Anam, MT.
Fajar Budiman, ST., M.Sc.

ABSTRACT

More than 100,000 large scale industries exist in Indonesia. In ASEAN, industry in Indonesia is the second largest country after Thailand. With a large number of industries, the number of PLN subscribers is increasing. Industrial customers are the biggest revenue pensupplay by PLN. One reuquired in industrial customers is the amount of usage. The device commonly used by this industry is resistive and inductive. In the inductive usage, the customer must pay attention to the power factor, the power factor can be assuredly caused by poor inductive quality. The nominal limit of power factor specified by PLN is 0.85. If the customer uses inductive less than 0.85 then it will be fined by PLN In this Final Project the tool made can measure the nominal VAR meter. In addition, if you see questions in the field about the vulnerability of customers subject to penalties by PLN due to inductive usage. Also equipped with media notification and recapitulation. With this inductive use can be monitored and supervised by the customer. Use of notification media with internet that integrated with user email and mobile phone number of customer. By sending sampling the waves read by the current and voltage sensors by through the number of samples and. Arduino cultivates waves and accumulates leading waves and laggings on waves and sizes of angular cosphi differences. In process of data retrieval process,error for current and voltage less than 3%, cosphi error angle less than 4%, and error for imaginary power less than 5%.

-----Halaman ini sengaja dikosongkan-----

KATA PENGANTAR

Dengan mengucap puji kami panjatkan kehadiran Allah Swt yang telah melimpahkan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan Tugas Akhir ini. Adapun judul Tugas Akhir ini yaitu

PERANCANGAN KVARH METER UNTUK METERING BEBAN INDUKTIF PADA JARINGAN TEGANGAN MENENGAH

Dalam Tugas Akhir ini dirancang system KVARH Meter dengan mencakup perhitungan cosphi, beban aktif, beban reaktif serta yang nantinya data akan diolah dengan internet. Tujuan penulisan Tugas Akhir ini dibuat sebagai salah satu syarat memperoleh gelar Ahli Madya (A. Md.) pada Program Studi Teknik Listrik, Departemen Teknik Elektro Otomasi, Institut Teknologi Sepuluh Nopember. Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, Tugas Akhir ini tidak dapat terselesaikan dengan baik. Oleh karenanya penulis ingin menyampaikan ucapan terimakasih kepada :

1. Kepala Departemen Teknik Elektro Otomasi Bapak Ir. Samsul Anam, MT. dan Bapak Fajar Budiman, ST., M.Sc. selaku Dosen Pembimbing atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya Tugas Akhir ini.
2. Staff/Karyawan/Dosen Departemen Teknik Elektro Otomasi.
3. Orang tua yang saya sayangi serta adik-adik dan juga sahabat dekat yang telah memberikan dukungan moral dan spiritual.
4. Rekan-rekan seperjuangan mahasiswa Departemen Teknik Elektro Otomasi angkatan 2014.

Penulis menyadari sepenuhnya bahwa Tugas Akhir ini jauh dari kata sempurna, oleh karena itu penulis mengharapkan saran dan kritik yang bersifat membangun untuk memperbaiki kekurangan dalam Tugas Akhir ini. Akhir kata penulis berharap semoga Tugas Akhir ini berguna bagi penulis pada khususnya dan bagi pembaca umumnya

Surabaya, 20 Juli 2017



Penulis

-----Halaman ini sengaja dikosongkan-----

DAFTAR ISI

HALAMAN

TUGAS AKHIR – TE 145561	i
FINAL PROJECT – TE 145561	iii
PERNYATAAN KEASLIAN.....	v
TUGAS AKHIR.....	vii
ABSTRAK.....	ix
<i>ABSTRACT</i>	xi
KATA PENGANTAR	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Permasalahan.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan	2
1.5 Sistematika Laporan	2
1.6 Relevansi	3
BAB II TEORI DASAR	
2.1 KVAR Meter	5
2.2 Board Arduino Mega.....	7
2.2.1 Komunikasi Serial	8
2.2.2 <i>Input dan Output</i>	9
2.2.3 Arduino IDE.....	9
2.3 Sensor Arus SCT013	10
2.4 <i>Real Time Clock DS1307 modules (RTC)</i>	11
2.5 <i>Liquid Crystal Display Keypad (LCD Keypad)</i>	13
2.6 <i>Module Data Logger</i>	14
2.7 <i>Miniature Circuit Breaker (MCB)</i>	15
BAB IIIPERANCANGAN DAN PEMBUATAN ALAT	
3.1 Blok Fungsional Sistem.....	17
3.2 Perancangan Mekanik	18
3.3 Perancangan Sensor Arus	21
3.4 Perancangan Sensor Tegangan	23
3.5 Perancangan <i>Wiring keypad LCD</i>	24
3.6 Perancangan <i>Wiring RTC</i>	24
3.7 Perancangan <i>Wiring Data Logger</i>	25
3.8 Pemrograman <i>Software Arduino IDE</i>	26
3.9 Pemrograman Sensor Tegangan	28

3.10 Pemrograman Sensor Arus	30
3.11 Pemrograman LCD	31
3.12 Pemrograman RTC	33
BAB IVPENGUJIAN DAN ANALISA DATA	
4.1 Pengujian <i>Power Supply</i>	37
4.2 Pengujian Arduino Mega 2560	39
4.3 Pengujian <i>Real Time Clock</i> (RTC).....	41
4.4 Pengujian LCD (<i>Liquid Crystal Display</i>).....	44
4.5 Pengujian <i>Keypad Module</i>	45
4.6 Pengujian Sensor Arus	46
4.7 Pengujian MCB.....	47
4.8 Pengujian Perhitungan VA.....	48
4.9 Pengujian Memori SD Card (Data Logger)	50
4.10 Pengujian Perhitungan Sensor Tegangan	52
4.11 Pengujian Perhitungan Cosphi	53
4.12 Pengujian Keseluruhan KVARH Meter	54
4.12Analisa Relevansi.....	59
BAB VPENUTUP	
5.1 Kesimpulan	61
5.2Saran	61
DAFTAR PUSTAKA	63

DAFTAR GAMBAR

	HALAMAN
Gambar 2.1	Penjumlahan Trigonometri Daya..... 6
Gambar 2.2	Arus Sephasa Dengan Tegangan 6
Gambar 2.3	Arus Tertinggal dari Tegangan Sebesar Sudut ϕ 7
Gambar 2.4	<i>Board</i> Arduino Mega 8
Gambar 2.5	Jendela Arduino IDE 10
Gambar 2.6	Sensor Arus SCT013 11
Gambar 2.7	<i>RTC Tiny I2C modules</i> 12
Gambar 2.8	Skematik RTC DS3231 12
Gambar 2.9	<i>Liquid Crystal Diplay Keypad</i> (LCD <i>Keypad</i>) 13
Gambar 2.10	<i>Wiring</i> Modul <i>MicroSD</i> Adapter 14
Gambar 2.11	<i>Mini Circuit Breaker</i> (MCB) 15
Gambar 0.1	Diagram Blok Fungsional Sistem 17
Gambar 3.2	<i>Wiring</i> Sistem Keseluruhan 18
Gambar 3.3	Perancangan Tata Letak Alat Satu 19
Gambar 3.4	Perancangan Tata Letak Alat Dua..... 20
Gambar 3.5	Desain Bentuk Depan KVARH Meter 21
Gambar 3.6	Rangkaian Pengkondisional Sensor Arus 22
Gambar 0.7	Perancangan Sensor Tegangan 23
Gambar 3.8	Perancangan <i>Wiring</i> LCD keypad 24
Gambar 3.9	<i>Wiring</i> RTC pada Arduino Mega..... 25
Gambar 3.10	<i>Wiring Data Logger</i> pada Arduino Mega 25
Gambar 3.11	<i>Flowchart</i> Pemrograman <i>Software</i> Arduino IDE..... 28
Gambar 3.12	<i>Flowchart</i> Pemrograman Sensor Tegangan 30
Gambar 3.13	<i>Flowchart</i> Pemrograman LCD <i>Keypad</i> 32
Gambar 3.14	<i>Flowchart</i> Pemrograman RTC..... 34
Gambar 3.15	<i>Flowchart</i> Pemrograman SD <i>Card</i> 36
Gambar 4.1	Proses Pengujian Arus <i>Power Supply</i> 38
Gambar 4.2	Proses Pengujian Tegangan <i>Power Supply</i> 38
Gambar 4.3	<i>Flowchart</i> Pemograman Pengujian Pada Pin 39
Gambar 4.4	<i>Flowchart</i> Pemograman untuk Pengujian RTC. 42
Gambar 4.5	Pengujian Data Waktu padaRTC Data 1 43
Gambar 4.6	Pengujian Data Waktu padaRTC Data 2 43
Gambar 4.7	Pengujian Data Waktu padaRTC Data 3 43
Gambar 4.8	Pengujian Tampilan LCD dengan Satu Baris..... 44

Gambar 4.9	Pengujian Tampilan LCD dengan Dua Baris.....	44
Gambar 4.10	Pengujian Tampilan <i>Push Button</i> Kondisi 1	45
Gambar 4.11	Pengujian Tampilan <i>Push Button</i> Kondisi 2	45
Gambar 4.12	Pengujian Perbandingan Hasil Sensor Arus	46
Gambar 4.13	Grafik Pembacaan Sensor Arus	47
Gambar 4.14	Pengujian MCB	49
Gambar 4.15	Perbandingan Pengukuran Daya	49
Gambar 4.16	Pengambilan Data Tampilan pada LCD	50
Gambar 4.17	Hasil Pengujian <i>SD Card</i> KVAR Meter.....	51
Gambar 4.18	<i>Flowchart</i> Penyimpanan Data .txt	52
Gambar 4.19	Hasil Pembacaan Sensor Tegangan.....	53
Gambar 4.20	Hasil Pembacaan Cosphi Meter.....	54
Gambar 4.21	Proses Pengambilan Cosphi pada Jenis Beban	56
Gambar 4.22	Interface pada UBIDOTS	57
Gambar 4.23	Proses Pengolahan Data pada UBIDOTS.....	57
Gambar 4.24	Tampilan Rekapitulasi Sudut Cosphi	58
Gambar 4.25	Tampilan Notifikasi via SMS Pelanggan	60
Gambar 4.26	Tampilan Notifikasi via Email Pelanggan	60

DAFTAR TABEL

HALAMAN

Tabel 2.1	Konfigurasi Pin LCD <i>Keypad</i> 16x2.....	14
Tabel 4.1	Vout dan Iout Dari <i>Power Supply</i>	39
Tabel 4.2	Pengujian Pin Arduino Kondisi Logika 1.....	40
Tabel 4.3	Pengujian Pin Arduino Kondisi Logika 0.....	41
Tabel 4.4	Pengujian Data Waktu padaRTC.....	42
Tabel 4.5	Data Pengambilan Data Arus	46
Tabel 4.6	Hasil Pengambilan Data Trip MCB 2 A.....	48
Tabel 4.7	Tabel Pengukuran Daya Semu	49
Tabel 4.8	Pengujian SD Card	51
Tabel 4.9	Pembacaan Sensor Tegangan	53
Tabel 4.10	Pembacaan Cosphi Meter	54
Tabel 4.11	Tabel Pembacaan Cosphi Beban	55

BAB I

PENDAHULUAN

1.1 Latar Belakang

Daya Listrik atau dalam bahasa Inggris disebut dengan Electrical Power adalah jumlah energi yang diserap atau dihasilkan dalam sebuah sirkuit/rangkaian. Sumber Energi seperti Tegangan listrik akan menghasilkan daya listrik sedangkan beban yang terhubung dengannya akan menyerap daya listrik tersebut. Dengan kata lain, Daya listrik adalah tingkat konsumsi energi dalam sebuah sirkuit atau rangkaian listrik. Jenis daya listrik dibagi menjadi tiga jenis, daya semu, daya aktif dan daya reaktif. Daya reaktif atau sering disebut VAR *volt ampere reactiv* merupakan suatu efek beban dari penggunaan beban induktif. daya reaktif adalah daya imajiner (khayal) yang menunjukkan adanya pergeseran arus dan tegangan listrik akibat adanya beban reaktif. Pengertian sederhananya daya reaktif adalah daya yang dibutuhkan sebagai pembangkit medan magnet di kumparan yang memiliki beban induktif. Contohnya adalah pada motor listrik induksi. Daya reaktif membangkitkan medan magnet di kumparan stator yang berfungsi untuk menginduksi rotor sehingga terciptalah medan magnet induksi di dalam komponen rotor tersebut. Beban reaktif tersebut bisa berupa beban induktif atau beban kapasitif.

Daya reaktif sangatlah diperhitungkan di dunia perlistrikan terutama di perindustrian. Besarnya beban induktif yang dihasilkan akan mempengaruhi besar kecilnya sudut $\cos\phi$. Jika dalam penggunaan beban sudut $\cos\phi$ nya kecil maka arus yang di hasilkan semakin besar. Penggunaan daya induktif yang membutuhkan hasil panas yang besar, jika beban menggunakan arus yang besar, maka rugi-rugi daya akan semakin besar. Dengan begitu PLN membuat peraturan sudut $\cos\phi$ penggunaannya harus kurang dari 0,85 yang dihitung secara terakumulatif dalam sebulan,

Permasalahan di atas yang mendasari diciptakannya karya berupa Sistem Notifikasi Prototype KVARH Meter via SMS dan E-mail. Alat ini lebih ditujukan kepada pelanggan yang dikenakan biaya KVARH oleh PLN. Pelanggan akan diberikan notifikasi apabila KVARH yang digunakan akan melebihi batas yang akan diakumulasi dalam sebulan. Dengan ini pelanggan dapat mengantisipasi agar nilai KVARH tidak melebihi batas yang sudah ditentukan. Selain itu, pelanggan juga tidak akan dikenakan denda.

1.2 Permasalahan

Beberapa masalah yang akan diselesaikan pada Perancangan KVARH Meter Untuk Metering Beban Induktif Pada Jaringan Tegangan Menengah pada jaringan tegangan menengah ini adalah:

Beban reaktif atau biasa disebut VAR merupakan jenis beban untuk membangkitkan suatu medan magnet. Beban berupa medan magnet ini biasa ditemui pada motor, rotor, transformator, atau beban dirumah seperti lampu TL. Beban reaktif ini biasanya tidak diperhitungkan untuk kalangan rumah tangga tetapi sangat diperhitungkan pada jaringan tegangan menengah atau pelanggan industri. Gelombang AC berbentuk sinus yang memiliki beda sudut antara arus dan tensinya. Perbedaan sudut *lagging* dan *leading* yang diakibatkan beban reaktif ini dibatasi oleh PLN sebesar cosphi 0,85. Penggunaan akumulasi cosphi yang tidak kurang dari 0,85 sangat dibutuhkan. Oleh karena itu pada Tugas Akhir prototype KVARH meter ini dilengkapi sistem notifikasi cosphi dan perhitungan segitiga daya secara lengkap dibutuhkan.

1.3 Batasan Masalah

Dari perumusan masalah di atas, ada beberapa hal yang perlu dibatasi sehingga penelitian yang dilakukan dapat tercapai. Batasan masalah dalam pengerjaan Tugas Akhir ini, yaitu:

- a. Pengukuran daya aktif dan reaktif pada jaringan satu fasa
- b. Akumulasi dan notifikasi penggunaan daya reaktif, daya aktif, dan daya semu
- c. Menentukan batasan pembacaan dan akumulasi cosphi

1.4 Tujuan

Tujuan dari Tugas Akhir ini adalah:

Tujuan dari Tugas Akhir ini adalah menghasilkan prototype KVARH meter dengan pengukuran daya aktif dan reaktif serta dilengkapi dengan notifikasi dan pengolahan data yang terintegrasi dengan pelanggan.

1.5 Sistematika Laporan

Pembahasan Tugas Akhir ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut:

Bab I Pendahuluan

Bab ini meliputi latar belakang, permasalahan, batasan masalah, tujuan penelitian, sistematika laporan, dan relevansi.

- Bab II Teori Dasar**
Bab ini menjelaskan tentang tinjauan pustaka, konsep dari KVARH meter
- Bab III Perancangan dan Pembuatan Alat**
Bab ini membahas perancangan baik secara program, mekanik dan elektronik untuk mengukur segitiga daya terutama daya reaktif.
- Bab IV Pengujian dan Analisa**
Bab ini memuat hasil pengujian serta analisa dari hasil pengujian tersebut.
- Bab V Penutup**
Bab ini berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

1.6 Relevansi

Hasil yang diperoleh dari Tugas Akhir ini diharapkan mampu menyelesaikan permasalahan pada pelanggan, diharapkan pelanggan dapat terbantu dengan terakumulasinya daya reaktif agar menghindari nominal batas cosphi minimal.

-----Halaman ini sengaja dikosongkan-----

BAB II TEORI DASAR

2.1 KVAR Meter

KVAR meter adalah alat untuk mengukur daya reaktif yang diserap suatu beban dalam waktu tertentu. Dalam penggunaannya KVAR meter lebih sering digunakan di dunia industri karena nilai faktor daya sangat berpengaruh terhadap penggunaan energi listrik. KVAR meter terdiri dari dua macam, yaitu:

a. Mekanik

Sistem kerjanya hampir sama dengan KWH meter yaitu perhitungannya berdasarkan perputaran cakram

b. Digital

Biasanya alat ukur ini menjadi satu dengan KVAH meter, KWH meter, serta cos phi meter

Sedangkan besarnya daya reaktif pada KVARh meter dapat dicari dengan rumus:

$$S = \sqrt{(P^2 + Q^2)} \dots\dots\dots(2.1)$$

Keterangan:

P = Daya Nyata (Watt) yaitu daya yang terpakai untuk melakukan energi sebenarnya dan dapat dicari dari rumus:

$$P = V \cdot I \cdot \cos \phi \dots\dots\dots(2.2)$$

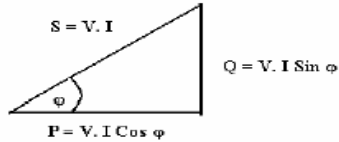
$$P = 3 \cdot V_L \cdot I_L \cdot \cos \phi$$

Q = Daya Reaktif (VAR) adalah jumlah daya yang diperlukan untuk pembentukan medan magnet dan dari pembentukan medan magnet maka akan terbentuk *fluks* medan magnet.

$$Q = V \cdot I \cdot \sin \phi \dots\dots\dots(2.3)$$

$$Q = 3 \cdot V_L \cdot I_L \cdot \sin \phi$$

S = Daya Kompleks (VA), adalah daya yang dihasilkan oleh perkalian antara tegangan rms dan arus rms dalam suatu jaringan atau daya yang merupakan hasil penjumlahan trigonometri daya aktif dan daya reaktif.



Gambar 2.1 Penjumlahan trigonometri daya aktif, reaktif dan semu

Dari Gambar 2.1 diatas terdapat faktor daya ($\cos \phi$) yang merupakan rasio perbandingan antara daya aktif (Watt) dan daya nyata (VA) yang digunakan dalam sirkuit AC atau beda sudut fasa antara V dan I .

Faktor daya yang dinotasikan sebagai $\cos \phi$ didefinisikan sebagai perbandingan antara arus yang dapat menghasilkan kerja didalam suatu rangkaian terhadap arus total yang masuk kedalam rangkaian atau dapat dikatakan sebagai perbandingan daya aktif (kW) dan daya semu (kVA). Daya reaktif yang tinggi akan meningkatkan sudut ini dan sebagai hasilnya faktor daya akan menjadi lebih rendah. Faktor daya selalu lebih kecil atau sama dengan satu. Dalam sistem tenaga listrik dikenal 3 jenis faktor daya yaitu faktor daya unity, faktor daya terbelakang (*lagging*) dan faktor daya terdahulu (*leading*) yang ditentukan oleh jenis beban yang ada pada sistem.

1. Faktor Daya *Unity*

Faktor daya unity adalah keadaan saat nilai $\cos \phi$ adalah satu dan tegangan sephasa dengan arus. Faktor daya *Unity* akan terjadi bila jenis beban adalah resistif murni



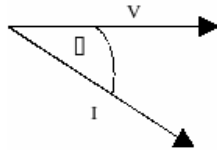
Gambar 2.2 Arus Sephase Dengan Tegangan

Pada Gambar 2.2 terlihat nilai $\cos \phi$ sama dengan 1, yang menyebabkan jumlah daya nyata yang dikonsumsi beban sama dengan daya semu.

2. Faktor Daya Terbelakang (*Lagging*)

Faktor daya terbelakang (*lagging*) adalah keadaan faktor daya saat memiliki kondisi-kondisi sebagai berikut :

1. Beban/ peralatan listrik memerlukan daya reaktif dari sistem atau beban bersifat induktif.
2. Arus (I) terbelakang dari tegangan (V), V mendahului I dengan sudut ϕ



Gambar 2.3 Arus tertinggal dari tegangan sebesar sudut ϕ

Dari Gambar 2.3 terlihat bahwa arus tertinggal dari tegangan maka daya reaktif mendahului daya semu, berarti beban membutuhkan atau menerima daya reaktif dari sistem. Beban/ peralatan listrik memberikan daya reaktif dari sistem atau beban bersifat kapasitif. Kemudian Arus mendahului tegangan, V terbelakang dari I dengan sudut ϕ

Penyebab faktor daya rendah yaitu diantaranya penggunaan beban induktif berupa Transformator, Motor induksi, Generator induksi, dan Lampu TL.

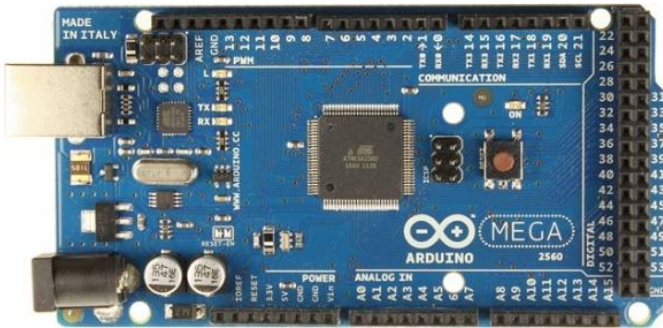
Alasan faktor daya diperbaiki untuk Mengurangi biaya pengoperasian peralatan listrik, Meningkatkan kapasitas sistem dan mengurangi rugi-rugi pada sistem yang dioperasikan, dan Mengurangi besarnya tegangan jatuh yang biasa disebabkan pada saat transmisi daya.

2.2 Board Arduino Mega

Board Arduino Mega adalah sebuah sistem minimum berbasis mikrokontroler ATmega2560. Spesifikasi board Arduino berdasarkan *datasheet* adalah sebagai berikut :

1. Spesifikasi Arduino Mega 2560 adalah sebagai berikut:
2. Menggunakan *chip microcontroller* AtMega2560.
3. Tegangan operasi 5 Volt.
4. Tegangan input (yang direkomendasikan, via *jack* DC) 7-12 Volt.
5. Digital I/O sebanyak 54 buah, 6 menyediakan PWM output.
6. Analog input pin sebanyak 16 buah.

7. Arus DC per pin I/O sebesar 20 mA.
8. Arus DC pada pin 3,3 Volt sebesar 50 mA.
9. *Flash memory* sebesar 256 KB, 8 KB telah digunakan untuk *bootloader*.
10. SRAM sebesar 8 kb.
11. EEPROM sebesar 4 kb.
12. *Clock speed* sebesar 16 Mhz.
13. Dimensi Arduino Mega 2560 sebesar 101,5 mm x 53,4 mm. 13. Berat Arduino Mega 2560 sebesar 37 g.



Gambar 2.4 Board Arduino Mega

Board ini dilengkapi *jack* DC dengan ukuran tegangan antara 6-20 V. Selain itu, *board* ini juga dilengkapi USB untuk komunikasi serial sehingga dapat dengan mudah dikoneksikan dengan PC. Bentuk fisik *board* Arduino seperti pada Gambar 2.4

2.2.1 Komunikasi Serial

Arduino Mega2560 memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, dengan Arduino lain, atau dengan mikrokontroler lainnya. Arduino ATmega328 menyediakan 4 *hardware* komunikasi serial UART TTL (5 Volt). Perangkat lunak Arduino termasuk didalamnya *serial monitor* memungkinkan data tekstual sederhana dikirim ke dan dari papan Arduino. LED RX dan TX yang tersedia pada papan akan berkedip ketika data sedang dikirim atau diterima melalui *chip* USB-to-serial yang terhubung melalui USB komputer (tetapi tidak untuk komunikasi serial seperti pada pin 0 dan 1).

2.2.2 Input dan Output

Masing-masing dari 54 digital pin pada Arduino Mega dapat digunakan sebagai *input* atau *output*, menggunakan fungsi *pinMode()*, *digitalWrite()*, dan *digitalRead()*. Arduino Mega beroperasi pada tegangan 5 Volt. Setiap pin dapat memberikan atau menerima arus maksimum 40 mA dan memiliki resistor *pull-up* internal (yang terputus secara *default*) sebesar 20-50 kOhms. Selain itu, beberapa pin memiliki fungsi khusus, antara lain:

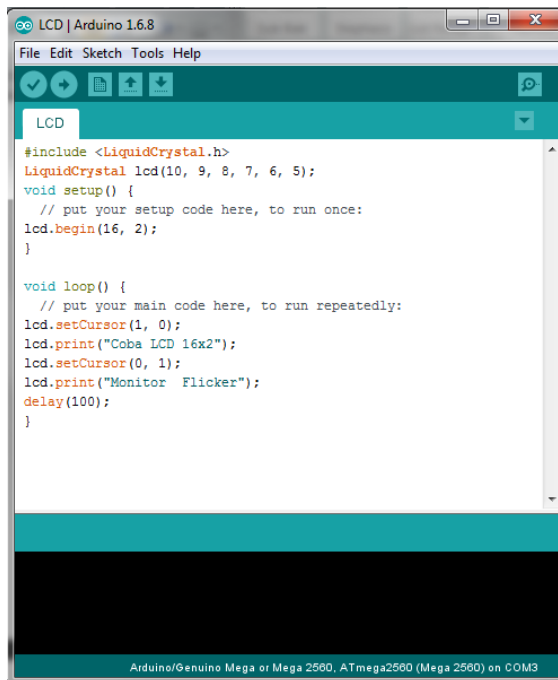
1. Serial 0 terdiri dari 0 (RX) dan 1 (TX), Serial 1 terdiri dari 19 (RX) dan 18 (TX), Serial 2 terdiri dari 17 (RX) dan 16 (TX), Serial 3 terdiri dari 15 (RX) dan 14 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) data serial TTL. Pins 0 dan 1 juga terhubung ke pin chip ATmega16U2 Serial USB-to-TTL.
2. Eksternal Interupsi : Pin 2 (*interrupt 0*), pin 3 (*interrupt 1*), pin 18 (*interrupt 5*), pin 19 (*interrupt 4*), pin 20 (*interrupt 3*), dan pin 21 (*interrupt 2*). Pin ini dapat dikonfigurasi untuk memicu sebuah interupsi pada nilai yang rendah, meningkat atau menurun, atau berubah nilai.
3. SPI : Pin 50 (MISO), pin 51 (MOSI), pin 52 (SCK), pin 53 (SS). Pin ini mendukung komunikasi SPI menggunakan perpustakaan SPI. Pin SPI juga terhubung dengan *header ICSP*, yang secara fisik kompatibel dengan Arduino Uno, Arduino Duemilanove, dan Arduino Diecimila.
4. LED : Pin 13. Tersedia secara *built-in* pada papan arduino ATmega2560. LED terhubung ke pin digital 13. Ketika pin diset bernilai high, maka LED menyala (on), dan ketika pin diset bernilai low, maka LED padam (off).
5. TWI : Pin 20 (SDA) dan pin 21 (SCL). Yang mendukung komunikasi TWI menggunakan perpustakaan *Wire*.

2.2.3 Arduino IDE

Board Arduino dapat di program menggunakan *software open source* bawaan Arduino IDE. Arduino IDE adalah sebuah aplikasi *crossplatform* yang berbasis bahasa pemrograman *Processing* dan *Wiring*. Arduino IDE di desain untuk mempermudah pemrograman dengan adanya kode editor yang dilengkapi dengan *syntax highlighting*, *brace matching*, dan indentasi otomatis untuk kemudahan pembacaan program, serta dapat meng-*compile* dan meng-*upload* program ke *board* dalam satu klik. Jendela Arduino IDE dapat dilihat pada Gambar 2.5 IDE Arduino adalah *software* yang sangat canggih ditulis dengan menggunakan Java. IDE Arduino terdiri dari:

Editor program merupakan sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*.

1. *Compiler* merupakan sebuah modul yang mengubah kode program (bahasa *Processing*) menjadi kode biner. Bagaimanapun sebuah *microcontroller* tidak akan bisa memahami bahasa *Processing*. Yang bisa dipahami oleh *microcontroller* adalah kode biner. Itulah sebabnya *compiler* diperlukan dalam hal ini.
2. *Uploader* merupakan sebuah modul yang memuat kode biner dari komputer ke dalam *memory* di dalam papan Arduino.



Gambar 2.5 Jendela Arduino IDE

2.3 Sensor Arus SCT013

Transformator arus (CT) adalah sensor yang mengukur arus bolak-balik. Trafo SCT013 merupakan trafo dengan jenis inti split, karena dapat dijepitkan langsung ke kabel bertegangan atau netral yang masuk pada beban yang akan diukur. Seperti trafo lainnya, transformator ini memiliki gulungan primer, inti magnetik, dan gulungan sekunder. Untuk

monitoring, gulungan primer digunakan untuk masuknya kabel bertegangan atau netral (hanya salah satu). Dan gulungan sekundernya disambungkan dengan alat pengukur (seperti rangkaian *microcontroller*).



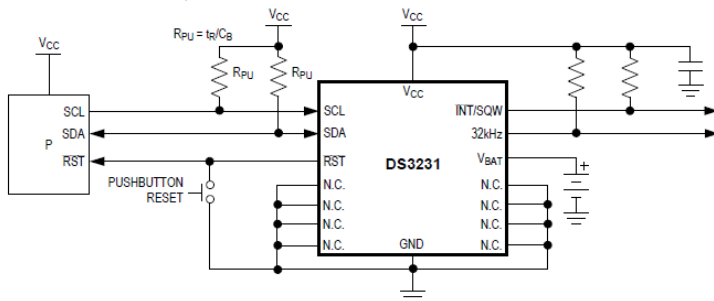
Gambar 2.6 Sensor Arus SCT013

Arus bolak-balik yang mengalir di primer menghasilkan medan magnet di inti, yang menginduksi arus di belitan sekunder. Arus dalam gulungan sekunder sebanding dengan arus yang mengalir dalam gulungan primer. Sensor arus yang digunakan adalah SCT013 seperti terlihat pada Gambar 2.6

2.4 Real Time Clock DS1307 modules (RTC)

Real Time Clock (RTC) adalah jenis pewaktu yang bekerja berdasarkan waktu yang sebenarnya atau dengan kata lain berdasarkan waktu yang ada pada jam kita. Meskipun istilah sering mengacu pada perangkat di komputer pribadi, *server* dan *embedded system*, RTC hadir di hampir semua perangkat elektronik yang perlu untuk menjaga keakuratan waktu. RTC memiliki sumber tenaga alternatif, sehingga mereka dapat terus menjaga waktu sementara sumber utama daya mati atau tidak tersedia. Sumber tenaga alternatif ini biasanya berupa baterai lithium dalam sistem lama, tetapi beberapa sistem yang lebih baru menggunakan supercapacitor, karena mereka dapat diisi ulang dan dapat disolder. Sumber daya alternatif juga dapat menyalurkan listrik ke RAM yang didukung baterai. Pada umumnya tenaga alternatif yang digunakan sebesar 3 Volt dari baterai lithium.

Gambar 2.7 RTC Tiny I2C *Modules*



Gambar 2.8 Skematik RTC DS3231

2.5 Liquid Crystal Diplay Keypad (LCD Keypad)

LCD (*Liquid Crystal Diplay*) berfungsi menampilkan suatu nilai hasil sensor, menampilkan teks, atau menampilkan menu pada aplikasimicrocontroller. Sumber cahaya di dalam sebuah perangkat LCD adalahlampu neon berwarna putih dibagian belakang susunan kristal cair tadi.

Titik cahaya yang jumlahnya puluhan ribu bahkan jutaan inilah yangmembentuk tampilan. Kutub kristal cair yang dilewati arus listrik akanberubah karena pengaruh polarisasi medan magnetik yang timbul danoleh karenanya akan hanya beberapa warna.

LCD membutuhkan driver supaya bisa dikoneksikan dengan sistem minimum dalam suatu microcontroller. Driver yang disebutkan berisi rangkaian pengaman, pengatur tingkat kecerahan maupun data, serta untuk mempermudah pemasangan di *microcontroller*.



Gambar 2.9 Liquid Crystal DiplayKeypad(LCDKeypad)

LCD Keypad dikembangkan untuk ArduinoShield, tujuannya untuk menyediakan antarmuka yang *user-friendly*dan memungkinkan pengguna untuk membuat berbagai pilihan menu dan lainnya. LCD Keypad ini terdiri dari 1602 karakter putih *backlight* biru. Seperti pada Gambar 2.5, terdapat 5 tombol yang terdiri dari *select*, *up*, *right*, *down* dan *left*. Untuk menyimpan pin IO digital, antarmuka keypad hanya

menggunakan satu saluran ADC. Berikut Tabel 2.1 adalah konfigurasi pin untuk LCD Keypad.

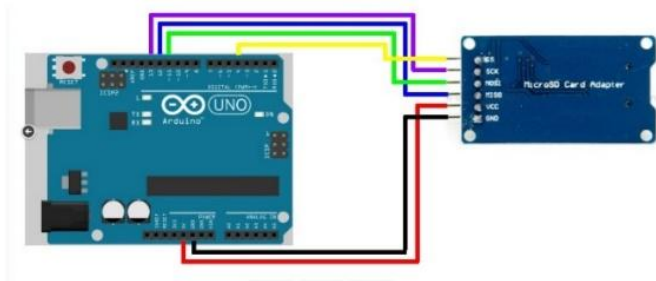
Tabel 2.1 Konfigurasi Pin LCD Keypad16x2

Pin	Function
Analog 0	Button (select, up, right, down and left)
Digital 4	DB4
Digital 5	DB5
Digital 6	DB6
Digital 7	DB7
Digital 8	RS (Data or Signal Display Selection)
Digital 9	Enable
Digital 10	Backlit Control

2.6 Module Data Logger

Pencatatan datanya diperlukan memori untuk menyimpan data yang telah terbaca. Memori yang digunakan adalah *microSD* yang telah terpasang pada modul *microSDAdapter*. Berikut adalah sambungan dan penjelasan gambar pada Gambar 2.9 antara *microSDAdapter* dengan Arduino Mega :

1. CS (*Chip Select*) – Pin 53 (SS) Arduino Mega
2. SCK (*Serial Clock*) – Pin 52 (SCK) Arduino Mega
3. MISO (*Serial Data Out*) – Pin 50 (MISO) Arduino Mega
4. MOSI (*Serial Data In*) – Pin 51 (MOSI) Arduino Mega
5. VCC – Pin 5 Volt Arduino Mega
6. GND – Pin *Ground* Arduino Mega



Gambar 2.10 Wiring Modul Microsdadapter dengan Arduino Mega

2.7 Miniature Circuit Breaker (MCB)

Mini Circuit Breaker (MCB) memiliki fungsi sebagai alat pengaman arus lebih. MCB ini memproteksi arus lebih yang disebabkan terjadinya beban lebih dan arus lebih karena adanya hubungan pendek. Prinsip dasar kerjanya yaitu untuk pemutusan hubungan yang disebabkan beban lebih dengan relai arus lebih sesaat menggunakan elektromagnet.

Bila elektromagnet bekerja, maka akan memutus hubungan kontak yang terletak pada pemadam busur dan membuka saklar. MCB untuk rumah seperti pada pengaman lebur diutamakan untuk proteksi hubungan pendek, sehingga pemakaiannya lebih diutamakan untuk mengamankan instalasi atau konduktornya. Arus nominal yang digunakan pada APP dengan mengenal tegangan 230/400 V ialah: 1 A, 2 A, 4 A, 6 A, 10 A, 16 A, 20 A, 25 A, 35 A, dan 50 A disesuaikan dengan tingkat VA konsumen. Adapun kemampuan membuka (breaking capacity) bila terjadi hubung singkat 3 KA dan 6 KA (SPLN 108-1993). MCB yang khusus digunakan oleh PLN mempunyai tombol biru. MCB pada saat sekarang ini paling banyak digunakan untuk instalasi rumah, instalasi industri maupun instalasi gedung bertingkat seperti pada Gambar 2.10



Gambar 2.11 *Mini Circuit Breaker* (MCB)

-----Halaman ini sengaja dikosongkan-----

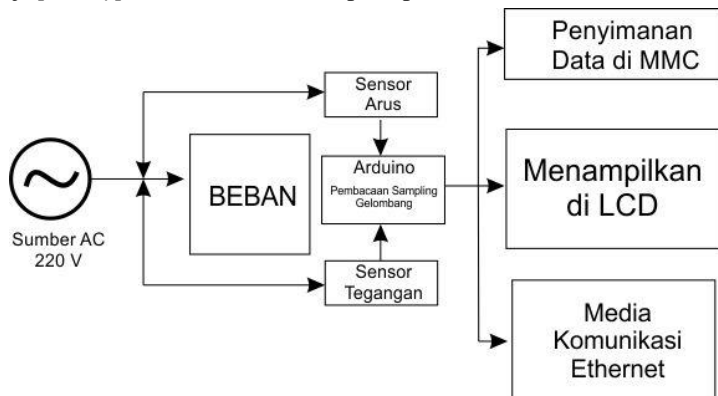
BAB III

PERANCANGAN DAN PEMBUATAN ALAT

Pada Bab ini akan dijelaskan mengenai Perancangan Kvarh Meter Untuk Metering Beban Induktif Pada Jaringan Tegangan Menengah meliputi, blok fungsional sistem yang akan menjelaskan proses kerja alat dalam bentuk alur diagram. Perancangan mekanik membahas tentang desain dan pembuatan mekanik yang mendukung cara kerja alat.

3.1 Blok Fungsional Sistem

Blok fungsional sistem ini menjelaskan tentang penjelasan system kerja *prototype* KVARH Meter. Seperti pada Gambar 3.1 berikut

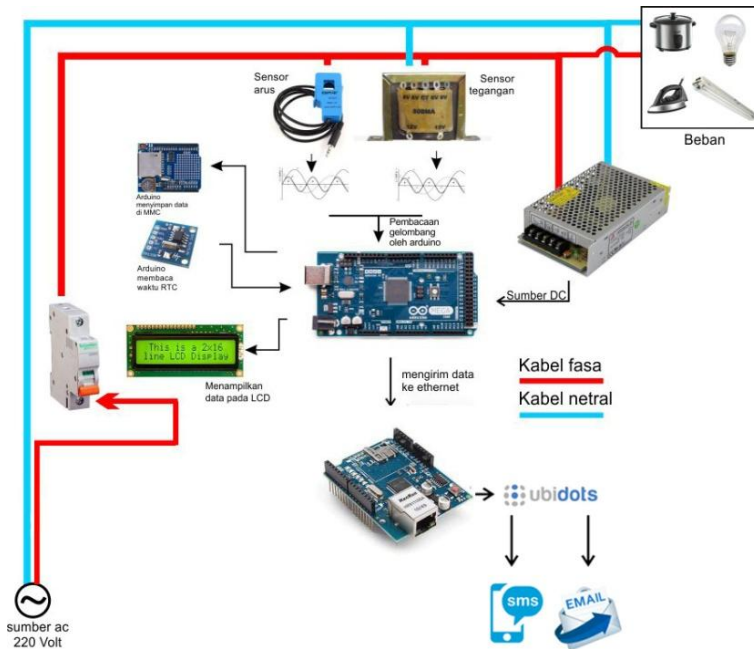


Gambar 3.1 Diagram Blok Fungsional Sistem

Terdapat beberapa komponen yang terhubung dengan Arduino Mega yaitu sensor arus, sensor tegangan, RTC, sebagai *input* serta LCD sebagai *output*. *Output* sensor arus dan tegangan terhubung pada pin *analog* Arduino Mega dan dibaca oleh ADC. Data ADC tersebut akan disimpan dalam besaran sesungguhnya pada program Arduino IDE setelah melalui proses linierisasi. Sensor arus dan tegangan terhubung pada pin *interrupt* eksternal Arduino Mega yang akan menjalankan fungsi *timer* sehingga diketahui selisih waktu antara gelombang arus dan tegangan, kemudian didapatkan nilai faktor dayanya. Setelah di temui faktor dayanya ditampilkan pada LCD. Data yang telah terekam pada alat akan di simpan pada MMC *data logger*. Kemudian data yang sduah ada pula di kirimkan ke modul Ethernet yang terintegrasi dengan IoT

3.2 Perancangan Mekanik

Perancangan mekanik meliputi perancangan elektronik dan tataletak baik saat perancangan dan pembuatan alat



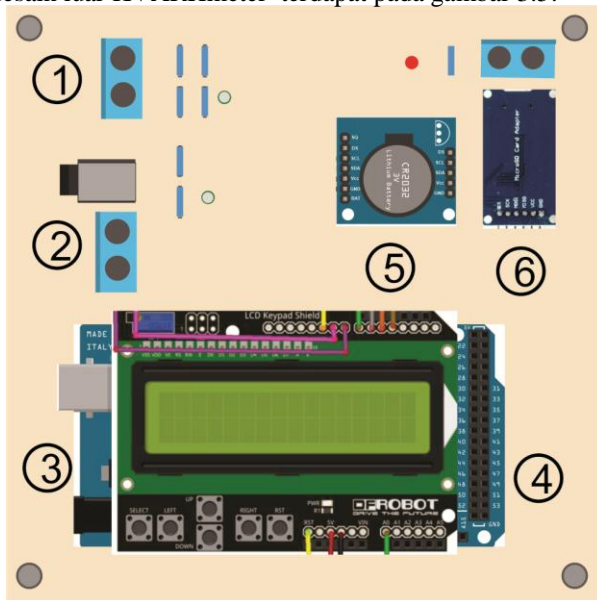
Gambar 3.2 Wiring Alat Keseluruhan

Pada bab ini akan dibahas mengenai perancangan dan pembuatan *prototype* KVARH meter tegangan menengah: perancangan dan pembuatan perangkat lunak (*software*) yang meliputi :

1. Perancangan Mekanik
2. Perancangan *Hardware* terdiri dari :
 - a Perancangan Sensor Tegangan
 - b Perancangan Sensor Arus
 - c Perancangan LCD
 - d Perancangan MCB
 - e Perancangan Rangkaian RTC
 - f Perancangan Rangkaian SD Card
 - g Perancangan Shield Arduino Mega

Perancangan mekanik berupa *prototype* KVAR Meter yang berada pada saluran distribusi pelanggan industri yang didalamnya terdapat

rangkaian *power supply*, *Arduino Shield*, modul ethernet, *LCD keypad*, rangkaian sensor tegangan, sensor arus, dan sensor *cos phi*. Tata letak perancangan mekanik terdapat pada gambar 3.3 dan 3.4 desain luar KVARHmeter terdapat pada gambar 3.5.

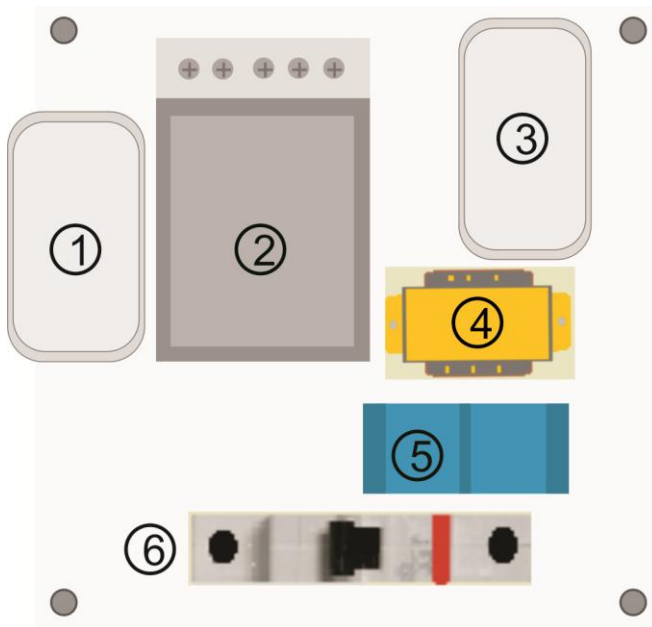


Gambar 3.3 Perancangan Tata Letak Alat Satu

Keterangan:

1. Sensor tegangan
2. Sensor Arus
3. Arduino Mega
4. LCD Keypad
5. RTC
6. Modul micro SD

Pada gambar dijelaskan tentang tata letak alat pada tingkatan wiring pertama yang meliputi sensor tegangan, sensor arus, Arduino Mega, LCD keypad, RTC, dan *Module Micro SD*

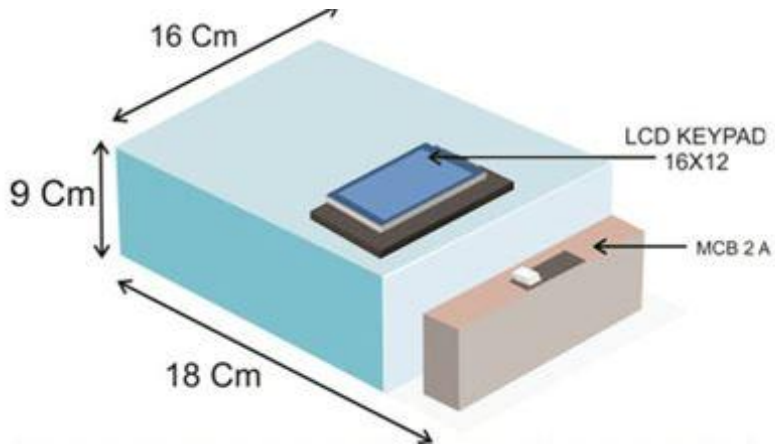


Gambar 3.4 Perancangan Tata Letak Alat Dua

Keterangan:

1. Terminal
2. *Power Supply*
3. Terminal
4. *Tranformator*
5. *Clamper arus*
6. MCB

Pada gambar dijelaskan tentang tata letak alat pada tingkatan wiring pertama yang meliputi terminal input, power supply 3A. Tranformator CT 350 mA, CT, dan alat pengaman berupa MCB.



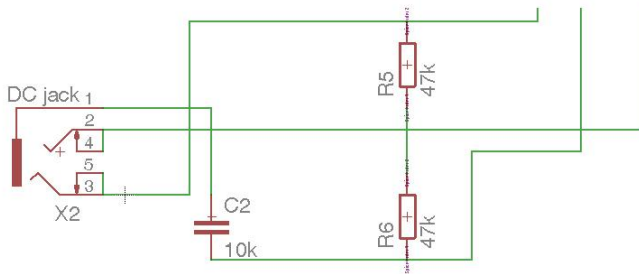
Gambar 3.5 Desain Bentuk Depan KVARH Meter

Keterangan :

1. LCD 16 x 12 digunakan untuk menampilkan informasi besarnya energi listrik, status konektivitas, informasi waktu, daya aktif, daya reaktif, dan daya imajiner
2. MCB 2 Ampere
3. Memiliki tinggi alat 9cm, lebar alat 16 cm, dan panjang alat 18 cm

3.3 Perancangan Sensor Arus

Sensor arus merupakan alat yang digunakan untuk mengetahui besarnya arus yang mengalir pada suatu peralatan listrik. Maka pada perancangan kami, digunakan sebuah transformator arus berupa CT YHDC SCT-013-000. YHDC SCT-013-000 yang digunakan dengan skala 30 A/1 V, penggunaannya dengan memasang *clamp* salah satu fasanya. Dengan menggunakan *clamp* ini langsung dapat mengukur tegangan AC langsung, dan nantinya akan dibaca dengan *output* analog menggunakan arduino.



Gambar 3.6 Rangkaian Pengkondisional Sensor Arus

Pada rangkaian sensor yang telah ditunjukkan pada Gambar 3.6 arus tersebut menggunakan DCjack untuk pembacaan hasil perhitungan analog dari YHDC SCT-013-000. Hasil pembacaan dari sensor arus tersebut dibaca oleh arduino berupa ADC. ADC tersebut dipotong oleh resistor. Tegangan 1 Volt dari output sensor masih memiliki tegangan negatif. Tegangan ini yang nantinya akan dimasukkan pada ADC arduino. Agar semua data dapat dibaca oleh arduino, nilai negatif pada tegangan harus diubah mejadi positif. Gelombang yang telah dipotong tersebut, dihilangkan bentuk noisenya dengan menggunakan kapasitor 10uF tersebut. Dengan begitu arduino dapat membaca nominal ADC dengan baik dan jelas.

Dengan sensor SCT13-10 yang memiliki Rasio Transformasi sebesar 1800 dan $R_{burden} = 62\Omega$, dapat diketahui besarnya arus yang ditransformasikan sensor sebelum menjadi tegangan adalah sebagai berikut :

$$I_{output\ CT} = \frac{I_{input}}{rasio} \dots\dots\dots (3.1)$$

$$I_{output\ CT} = \frac{30\ A}{1800}$$

$$I_{output\ CT} = 0,0166\ A$$

Tegangan yang dihasilkan oleh sensor arus setelah melewati resistor *burden* adalah sebesar :

$$V_{CT} = R_{burden} \times Arus_{CT} \dots\dots\dots (3.2)$$

$$V_{CT} = 62\Omega \times 0,0166\ A$$

$$V_{CT} = 1,02\ Volt\ (1\ Volt)$$

Untuk mengubah tegangan negatif menjadi tegangan positif dengan menggunakan rangkaian clamper yang akan meng

geser sinyal. Pertama dengan membuat tegangan *mid-point* dengan menggunakan 2 resistor yang sama besar :

$$V_{mid} = \frac{R1}{R1+R2} \times V_{in \text{ mid}} \dots \dots \dots (3.3)$$

$$V_{mid} = \frac{10K}{10K+10K} \times 5 \text{ Volt}$$

$$V_{mid} = 2,5 \text{ Volt}$$

Dengan begitu dapat disimpulkan bahwa tegangan yang masuk ke ADC Arduino Mega 2560 adalah sebagai berikut:

$$V_{pick \text{ to pick}} = (2 \times V_{out}) + V_{mid} \dots \dots \dots (3.4)$$

$$V_{pick \text{ to pick}} = (2 \times 1) + 2,5 \text{ Volt}$$

$$V_{pick \text{ to pick}} = 4,5 \text{ Volt}$$

$$V_{pmax} = V_{mid} + V_{out} \dots \dots \dots (3.5)$$

$$V_{pmax} = 2,5 \text{ Volt} + 1 \text{ Volt}$$

$$V_{pmax} = 3,5 \text{ Volt}$$

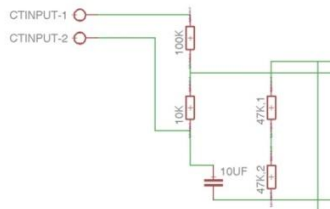
$$V_{pmin} = V_{mid} - V_{out} \dots \dots \dots (3.6)$$

$$V_{pmin} = 1,5 \text{ Volt}$$

3.4Perancangan Sensor Tegangan

Tugas akhir ini menggunakan sensor tegangan dari *transformer step down* 220 Volt / 12 Volt. Trafo *stepdown* digunakan untuk menurunkan tegangan 220 Volt agar nantinya dapat dibaca oleh ADC pada arduino yang membutuhkan tegangan 0-5 Volt.

Agar hasil dari sensor tegangan dapat dibaca oleh ADC arduino maka dilakukan pembagian tegangan agar didapat nilai tegangan yang lebih kecil. Tegangan yang akan masuk pada ADC masih berupa gelombang sinus yang memiliki nilai negatif. Agar semua data dapat dibaca oleh arduino, nilai negatif pada tegangan harus diubah mejadi positif. Berikut *wiring* sensor tegangan yang terapat pada Gambar 3.7



Gambar 3.7 Perancangan Sensor Tegangan

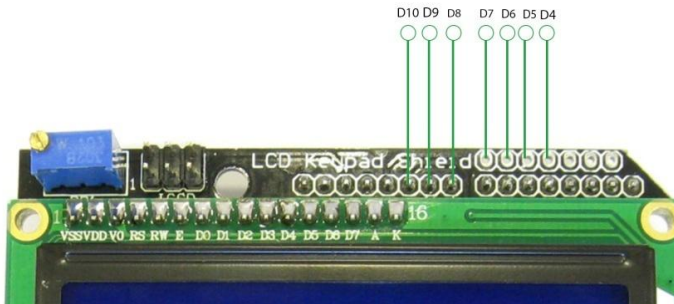
3.5 Perancangan Wiring keypad LCD

Liquid Crystal Display atau LCD yang terdapat pada Gambar 3.7 digunakan untuk mempermudah *user* untuk melihat kondisi terkini dari relai. LCD yang digunakan adalah LCD Keypad 16 x 2. Menu yang akan ditampilkan ada 3, yaitu :

1. Menu *Display* yang berisi *I setting*, *I rms* dan *V rms*
2. Menu Faktor daya, *KVAR*, *KW*, dan *VAR*

Berikut adalah pengkabelan yang harus dilakukan untuk menghubungkan LCD Keypad dengan Arduino Mega 2560 :

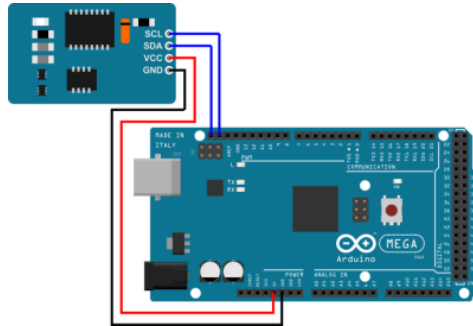
1. Pin A0 dengan pin 5 tombol LCD Keypad
2. Pin Digital 4 dihubungkan dengan pin DB 4
3. Pin Digital 5 dihubungkan dengan DB5
4. Pin Digital 6 dihubungkan dengan DB 6
5. Pin Digital 7 dihubungkan dengan DB7
6. Pin Digital 8 dihubungkan dengan pin RS
7. Pin Digital 9 dihubungkan dengan pin *Enable*
8. Pin Digital 10 dihubungkan dengan pin *Backlight Control*



Gambar 3.8 Perancangan Wiring LCD Keypad pada Arduino Mega

3.6 Perancangan Wiring RTC

Modul RTC menggunakan komunikasi I2C untuk bertukar data dengan Arduino Mega. Modul ini memiliki 4 pin yang harus terhubung dengan Arduino Mega. Wiring hubungan antara RTC dengan Arduino Mega dapat dilihat pada Gambar 3.9



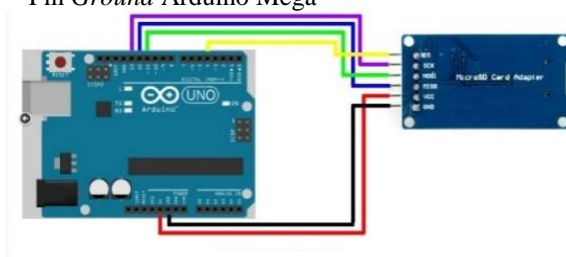
Gambar 3.9 Wiring RTC pada Arduino Mega

1. Pin VCC terhubung dengan referensi tegangan 5 volt Arduino Mega
2. Pin GND terhubung dengan *ground* Arduino Mega
3. Pin SDA terhubung dengan pin SDA Arduino Mega
4. Pin SCL terhubung dengan pin SCL Arduino Mega

3.7 Perancangan *WiringData Logger*

Untuk pencatatan datanya diperlukan memori untuk menyimpan data yang telah terbaca. Memori yang digunakan adalah *microSD* yang telah terpasang pada modul *microSDAdapter*. Gambar 3.8 berikut adalah sambungan antara *microSDAdapter* dengan Arduino Mega :

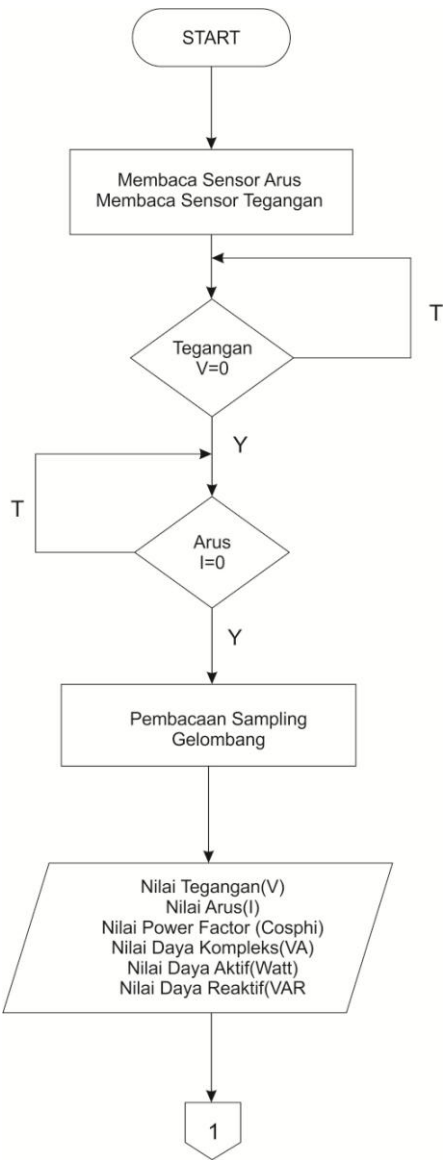
1. CS (*Chip Select*) – Pin 53 (SS) Arduino Mega
2. SCK (*Serial Clock*) – Pin 52 (SCK) Arduino Mega
3. MISO (*Serial Data Out*) – Pin 50 (MISO) Arduino Mega
4. MOSI (*Serial Data In*) – Pin 51 (MOSI) Arduino Mega
5. VCC – Pin 5 Volt Arduino Mega
6. GND – Pin *Ground* Arduino Mega

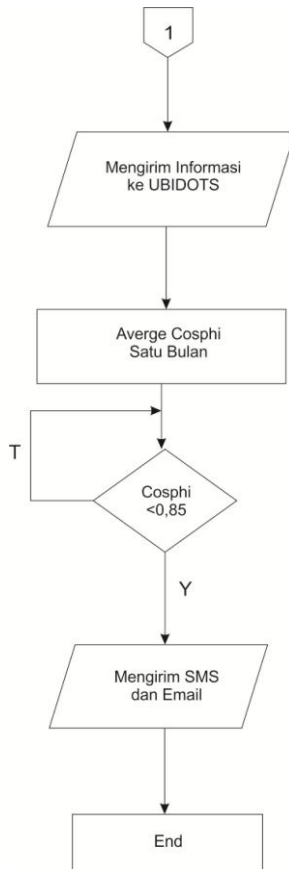


Gambar 3.10 Wiring Data Logger pada Arduino Mega

3.8 Pemrograman *Software* Arduino IDE

Software Arduino IDE pada Tugas Akhir digunakan untuk melakukan pemrograman papan Arduino dalam menjalankan sistem secara keseluruhan.





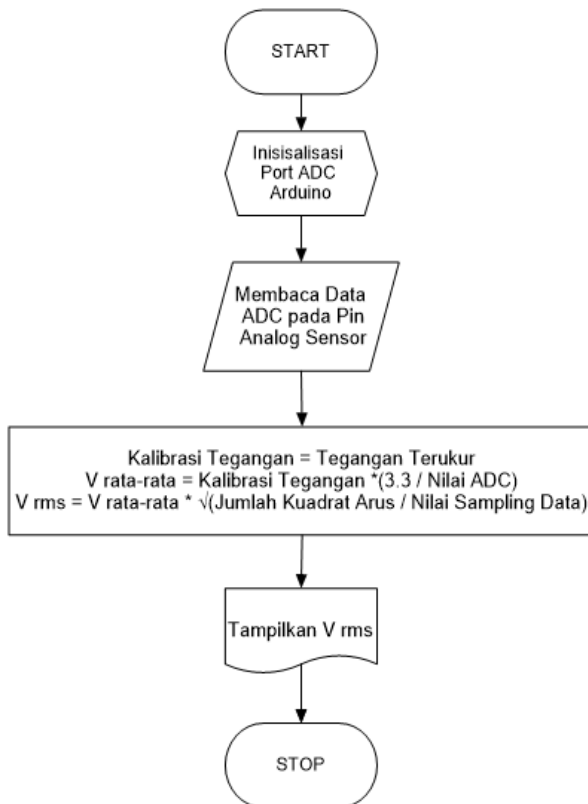
Gambar 3.11 Flowchart Pemrograman Software Arduino IDE

Pada alur flowchart yang terdapat pada Gambar 3.11 dapat dilihat bahwa program bekerja dengan pembacaan nilai arus dan tegangan yang terbaca oleh sensor. Setelah nilai arus dan tegangan terbaca, arduino akan membaca sampling gelombang yang terbaca. Setelah sampling gelombang tersebut terbaca oleh arduino, akan diolah dan menampilkan menjadi beberapa hasil yaitu nilai arus, nilai tegangan, nilai cosphi, nilai daya reaktif, dan jumlah pemakaian daya. Kemudian hasil dari data yang telah ditampilkan dan diolah data dikirim ke IoT yaitu UBIDOTS. Data yang telah terekam pada UBIDOTS, akan diolah hasil rata-rata pemakaian dalam sebulannya. Jika ditemui cosphi dengan nilai dibawah

0,85 program akan memberikan perintah untuk mengirim notifikasi berupa sms dan email kepada pelanggan. Program ini berlangsung loop terus menerus.

3.9 Pemrograman Sensor Tegangan

Sensor tegangan yang digunakan pada Tugas Akhir ini menggunakan trafo CT 350 mA dengan tegangan 220 Volt / 9 Volt. Dimana *output* dari trafo ini berupa tegangan 9 Volt yang memiliki siklus negatif. Dengan konversi yang telah dijelaskan pada perancangan *hardware* maka akan didapat tegangan yang dapat terbaca oleh Arduino. Agar sensor tegangan ini dapat digunakan sebagai sensor, dibutuhkan program yang sesuai untuk sensor ini. *Flowchart* untuk program sensor tegangan



Gambar 3.12 *Flowchart* Pemrograman Sensor Tegangan

Digambarkan pada Gambar 3.12 *Flowchart* diatas, untuk urutan cara kerja dari *flowchart* adalah sebagai berikut :

1. *Start* adalah ketika program dimulai.
2. Untuk *wiring* Sensor Tegangan dengan Arduino dihubungkan dengan pin analog Arduino atau pin ADC
3. Selanjutnya inisialisasi variabel yang akan digunakan untuk pembacaan dan penghitungan tegangan.
4. Selanjutnya membaca data pada analog Arduino atau ADC
5. Dengan menggunakan perhitungan sebagai berikut akan memperoleh nilai V_{rms} :

Kalibrasi Tegangan = Tegangan Terukur

$$V_{rata-rata} = \text{Kalibrasi Tegangan} * \frac{3.3}{\text{Nilai ADC}}$$

$$V_{rms} = V_{rata-rata} * \sqrt{\frac{\text{Jumlah Kuadrat Arus}}{\text{Nilai Sampling Data}}}$$

6. Menampilkan nilai V_{rms} dalam serial monitor

Untuk mengubah tegangan negatif menjadi tegangan positif dengan menggunakan rangkaian clamper yang akan menggeser sinyal. Pertama dengan membuat tegangan mid-point dengan menggunakan 2 resistor yang sama besar :

$$V_{mid} = \frac{R1}{R1+R2} \times V_{in \ mid} \dots\dots\dots(3.7)$$

$$V_{mid} = \frac{470k}{470k+470k} \times 5 \text{ Volt}$$

$$V_{mid} = 2,5 \text{ Volt}$$

Dengan begitu dapat disimpulkan bahwa tegangan yang masuk ke ADC Arduino Mega 2560 adalah sebagai berikut:

$$V_{pick \ to \ pick} = (2 \times V_{out}) + V_{mid} \dots\dots\dots(3.8)$$

$$V_{pick \ to \ pick} = (2 \times 1,09) + 2,5 \text{ Volt}$$

$$V_{pick \ to \ pick} = 4,68 \text{ Volt}$$

$$V_{pmax} = V_{mid} + V_{out} \dots\dots\dots(3.9)$$

$$V_{pmax} = 2,5 \text{ Volt} + 1,09 \text{ Volt}$$

$$V_{pmax} = 3,59 \text{ Volt}$$

$$V_{pmin} = V_{mid} - V_{out} \dots\dots\dots(3.10)$$

$$V_{pmin} = 2,5 \text{ Volt} - 1,09 \text{ Volt}$$

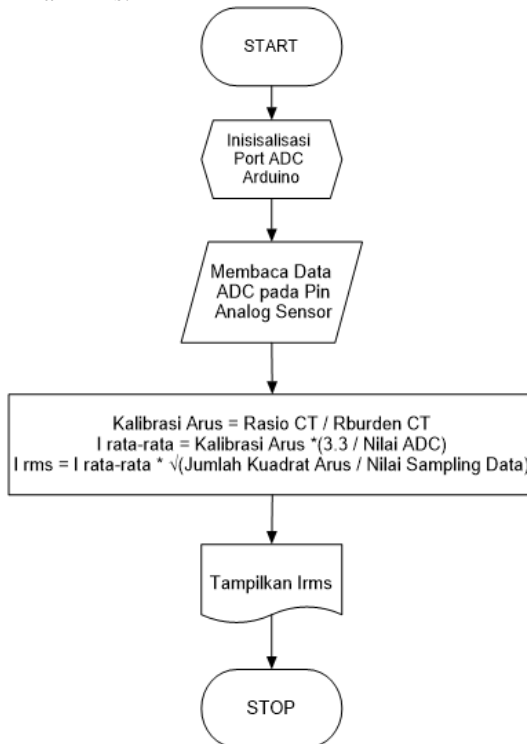
$$V_{pmin} = 1,41 \text{ Volt}$$

3.10 Pemrograman Sensor Arus

Sensor arus yang digunakan pada Tugas Akhir ini adalah sensor arus *current transformers* SCT013-10 dimana *output* dari sensor ini berupa tegangan dan yang nantinya harus dapat terbaca oleh Arduino. Sensor arus ini dapat digunakan sebagai sensor berikut langkah perancangan sensor arus, burden yang terdapat pada SCT013 sebesar 68k. Pada pola pemrograman dibutuhkan sistem kalibrasi yang tepat untuk mengukur arus yang tepat dari beban. Nominal kalibrasi tersebut agar dapat dibaca oleh arduino dengan ADC dalam nominal Irms dari keluaran SCT013. Berikut flowchart dari pemrograman sensor arus:

$$I_{rms} = I_{rata-rata} * \sqrt{\frac{\text{Jumlah Kuadrat Arus}}{\text{Nilai Sampling Data}}}$$

Menampilkan nilai I rms:



Gambar 3.13 Flowchart Pemrograman Sensor Arus SCT-013

Seperti yang telah digambarkan pada Gambar 3.13 *Flowchart* diatas, untuk urutan cara kerja dari *flowchart* adalah sebagai berikut :

1. *Start* adalah ketika program dimulai.
2. Untuk *wiring* Sensor Arus dengan Arduino dihubungkan dengan pin analog Arduino
3. Selanjutnya inisialisasi variabel yang akan digunakan untuk pembacaan dan penghitungan arus.
4. Selanjutnya membaca data pada analog Arduino atau ADC
5. Dengan menggunakan perhitungan sebagai berikut akan memperoleh nilai I rms :

$$\text{Kalibrasi Arus} = \text{Rasio CT} / \text{Rburden CT} \dots\dots\dots(3.11)$$

$$\text{I rata-rata} = \text{Kalibrasi Arus} * \frac{3.3}{\text{Nilai ADC}} \dots\dots\dots(3.16)$$

$$\text{I rms} = \text{I rata-rata} * \sqrt{\frac{\text{Jumlah Kuadrat Arus}}{\text{Nilai Sampling Data}}} \dots\dots\dots(3.17)$$

6. Menampilkan nilai I rms

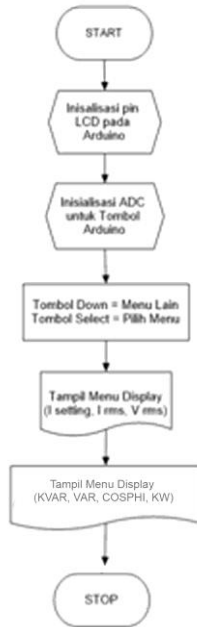
3.11 Pemrograman LCD

Liquid Crystal Display atau LCD digunakan untuk mempermudah *user* untuk melihat kondisi terkini dari relai. LCD yang digunakan adalah LCD *Keypad* 16 x 2. Pada pemrograman pada Arduino IDE telah tersedia *library* pemrograman LCD. Untuk *keypad* dan menu maka dibuat program tersendiri untuk melengkapi program yang telah ada.

Agar LCD *Keypad* ini dapat digunakan sebagai penampil menu kondisi yang terjadi pada relai maka dibutuhkan program yang sesuai untuk LCD *Keypad* ini. Untuk urutan cara kerja dari *flowchart* adalah sebagai berikut :

1. *Start* adalah ketika program dimulai.
2. Untuk *wiring* LCD *Keypad* ini tombol dihubungkan pada Analog 0 Arduino, DB 4 pada Digital 4, DB5 pada digital 5, DB 6 pada Digital 6, DB7 pada Digital 7, pin RS pada Digital 8, *Enable* pada Digital 9 dan *Backlight Control* pada pin Digital 10.
3. Selanjutnya inisialisasi ADC yang akan digunakan untuk mengatur tombol pada LCD *Keypad*.
4. Tombol yang digunakan adalah tombol *Down* untuk menampilkan pilihan menu dan tombol *Select* untuk memilih menu.

5. Menu yang akan ditampilkan ada 3, yaitu :
 - a) Menu *Display* yang berisi *I setting*, *I rms* dan *V rms*
 - b) Menu Faktor daya, KVAR, KW, dan VAR
6. *Stop* adalah ketika program berhenti.



Gambar 3.14 *Flowchart* Pemrograman LCD Keypad

Untuk memudahkan dalam memahami *flowchart* pada Gambar 3.14 di atas, maka akan dijelaskan pada tahap-tahap berikut ini.

TAHAP 1

Pada tahap ini dilakukan inisialisasi dan konfigurasi pada setiap input dan output pada Arduino Mega. Terdapat lima *input* yang terpasang pada Arduino Mega yaitu :

1. Sensor arus menggunakan pin *analog* A1
2. Sensor tegangan menggunakan pin *analog* A2
3. Sensor *cos phi* menggunakan pin *interrupt* eksternal D2 dan D3

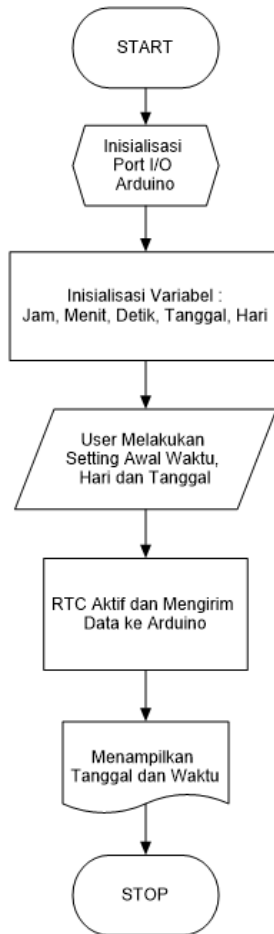
4. RTC (*Real Time Clock*) menggunakan komunikasi I2C pin SDA dan SCL.
5. Keypad menggunakan pin digital D38, D40, D42, D44, D46, D48, D50, D52.
Sedangkan output yang terpasang sebanyak dua, yaitu:
6. LCD (*Liquid Crystal Display*) menggunakan pin digital D43, D45, D47, D49, D51, D53.
antara Arduino Mega dengan Arduino Uno. Serial1 menggunakan pin D18, dan D19.

3.12 Pemrograman RTC

Real Time Clock yang digunakan pada Tugas Akhir ini, yaitu Modul *Real Time Clock* DS1307. Untuk pemrograman pada papan Arduino telah tersedia *library* dari RTC tersebut. *Wiring* RTC ke Arduino dengan I₂C, yaitu menghubungkan pin SDA dan SCL dari RTC ke Arduino pada pin 20 dan 21. Pengguna dapat memanggil *library* yang telah tersedia pada Arduino IDE.

Agar RTC ini dapat digunakan sebagai pemberi data waktu dengan baik maka dibutuhkan program yang sesuai untuk RTC ini. Pada RTC ini set awal untuk data hari, tanggal, serta waktu diberikan pada program dan tidak di set secara manual setelah program diupload. Untuk urutan cara kerja dari *flowchart* adalah sebagai berikut :

1. *Start* adalah ketika program dimulai.
2. Untuk *wiring* RTC dengan Arduino dihubungkan pada pin SDA dan SCL jadi pada inisialisasi *port* I/O Arduino harus mengaktifkan pin SDA dan SCL. Yakni dengan mengaktifkan komunikasi I₂C.
3. Selanjutnya inisialisasi variabel yang akan digunakan untuk detik, menit, jam, tanggal, dan hari.
4. Selanjutnya *user* akan melakukan *setting* awal untuk waktu pada RTC tersebut.
5. Dengan data *setting* awal tersebut RTC akan mengirim data tersebut ke Arduino dengan komunikasi I₂C yakni melalui pin SDA dan SCL.
6. Data waktu akan ditampilkan pada LCD.



Gambar 3.14 *Flowchart* Pemrograman RTC

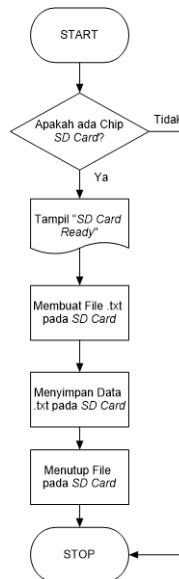
3.13 Pemrograman SD Card

Pemrograman data *logger* juga menggunakan pemrograman Arduino IDE. Komunikasi data *logger* berbeda dengan komunikasi dengan RTC yaitu menggunakan SPI. Lebih jelasnya menggunakan pin MISO, MOSI, SCK, dan CS. Pada rancangan *hardware* pin CS terletak pada pin I/O 53. Pin tersebut akan digunakan sebagai acuan untuk mengaktifkan komunikasi dengan relai. Agar *SD Card* ini berjalan

dengan baik pada Arduino, maka dibutuhkan pemrograman yang sesuai dengan kebutuhan dari *SD Card*. Untuk urutan cara kerja dari *flowchart* pada Gambar 3.15 adalah sebagai berikut :

1. *Start* adalah ketika program dimulai.
2. Dilakukan pengecekan untuk mengetahui ada atau tidaknya *chip SD Card*.
3. Jika terdapat *chip SD Card* maka akan dimulai untuk proses penyimpanan dari data yang akan disimpan, sedangkan jika tidak ada *chip SD Card* maka program akan berhenti.
4. Tampilan “*SD Card Ready*” untuk proses selanjutnya.
5. Pada tahap ini akan dimulai proses penyimpanan data yang diawali dengan membuka *file* pada *SD Card*.
6. Data akan tersimpan pada *file* yang telah dibuka pada tahap sebelumnya.
7. Setelah data tersimpan, selanjutnya *file* akan ditutup dan data telah selesai tersimpan

Program tersebut digunakan untuk menulis data *String* pada *file*. Setiap pengaksesan *SD Card* dimulai dengan perintah `SD.open()`; untuk menulis data yang tersimpan dalam memori pada *file* menggunakan `dataFile.print ()`; ketika penulisan selesai maka akan ditutup dengan `dataFile.close()`;



Gambar 3.15 *Flowchart* Pemrograman *SD Card*

-----Halaman ini sengaja dikosongkan-----

BAB IV

PENGUJIAN DAN ANALISA DATA

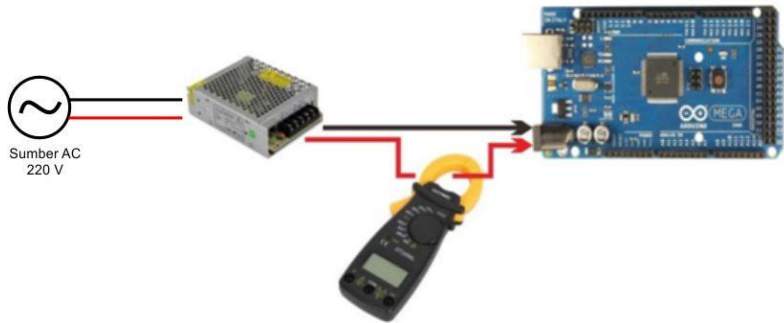
Pada bab ini akan membahas mengenai hasil pengujian dan analisa atas penyusunan alat untuk “Perancangan KVARH Meter Untuk Metering Beban Induktif Pada Jaringan Tegangan Menengah “. Data pengujian sangat diperlukan untuk implementasi dalam dunia nyata. Kinerja suatu sistem sangat dipengaruhi oleh kinerja per bagian dari sistem tersebut seperti

Pengujian merupakan salah satu langkah yang harus dilakukan untuk mengetahui apakah sistem yang telah dibuat sesuai dengan yang direncanakan. Pengujian juga bertujuan untuk mengetahui kelebihan dan kekurangan dari sistem yang telah dibuat. Hasil pengujian ini akan dianalisa untuk mengetahui penyebab terjadinya kekurangan atau kesalahan dalam sistem. Pengujian tersebut meliputi :

1. Pengujian *Power Supply*
2. Pengujian Arduino Mega 2560
3. Pengujian *Real Time Clock (RTC)*
4. Pengujian LCD (*Liquid Crystal Display*)
5. Pengujian *Keypad Module*
6. Pengujian MCB
7. Pengujian Datal Logger Module
8. Pengujian Sensor Tegangan
9. Pengujian Sensor Arus
10. Pengujian VAR
11. Pengujian Cosphi
12. Pengujian Keseluruhan

4.1 Pengujian *Power Supply*

Pengujian *power supply* terdiri dari pengujian tegangan *input power supply* dan pengujian tegangan *output power supply*. Nameplate yang tertera pada *power supply* yaitu mempunyai tegangan sebesar 12 VDC dengan arus 3 Ampere. Pengujian ini dilakukan untuk mengetahui besarnya tegangan *input* pada rangkaian *power supply*. Pengujian ini dilakukan pada *power supply* yang akan di hubungkan pada *board* Arduino guna pengujian arus yang memberikan *supply* pada Arduino, seperti yang terlihat pada gambar 4.1. Pengujian ini menggunakan *multimeter* “SANWA” yang bertujuan untuk menguji apakah tegangan sumber dan arus yang akan digunakan sudah sesuai dengan keperluan Arduino . Untuk skema proses pengujian tegangan *power supply* dapat dilihat pada Gambar 4.1 dan Gambar 4.2.



Gambar 4.1 Proses Pengujian Arus *Power Supply* dengan Arduino



Gambar 4.2 Proses Pengujian Tegangan *Power Supply*

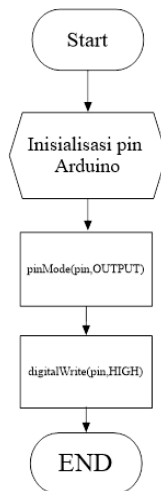
Tabel 4.1 Vout dan Iout Dari Power Supplay

Vout	Iout
12.09 Volt	0.9 A
9.02 Volt	0.8 A
7.14 Volt	0.6 A
5.49 Volt	0.4 A
3.72 Volt	0.3 A

Dari data pada Tabel 4.1 bahwa tegangan keluaran *power supply* 12.09 Volt. Nilai tegangan pada DC female mikrokontroler atmega 2560 mempunyai *range* kerja yaitu 9-12 Volt. Arus yang terukur sendiri tidak melebihi dari 2,5 Ampere. Sehingga tegangan dan arus *powersupply* masih dapat digunakan untuk memberikan *supply* ke rangkaian Arduino.

4.2 Pengujian Arduino Mega 2560

Pengujian ini dilakukan terhadap *board* Arduino Mega 2560. Pengujian dilakukan pada pin yang digunakan pada sistem alat ini yaitu pin A0 s/d A7, D0 s/d D21 milik Arduino Mega 2560 dengan cara mengatur keluarannya pada logika 1 dan 0 dengan mengunggah program ke *board* Arduino. Flowchart program arduino dapat dilihat pada Gambar 4.3.



Gambar 4.3 Flowchart Pemograman Pengujian Pada Pin Arduino

Tabel 4.2 Pengujian Pin Arduino Kondisi Logika 1

No	Pin dan Status	V Output
1	digitalWrite(A0,HIGH);	5,1
2	digitalWrite(A1,HIGH);	5,1
3	digitalWrite(A2,HIGH);	5,1
4	digitalWrite(A3,HIGH);	5,1
5	digitalWrite(A4,HIGH);	5,2
6	digitalWrite(A5,HIGH);	5,1
7	digitalWrite(A6,HIGH);	5,1
9	digitalWrite(A7,HIGH);	5,2
10	digitalWrite(0,HIGH);	5,2
11	digitalWrite(1,HIGH);	5,1
12	digitalWrite(2,HIGH);	5,1
13	digitalWrite(3,HIGH);	5,1
14	digitalWrite(4,HIGH);	5,1
15	digitalWrite(5,HIGH);	5,1
16	digitalWrite(6,HIGH);	5,1
17	digitalWrite(7,HIGH);	5,2
18	digitalWrite(8,HIGH);	5,1
19	digitalWrite(9,HIGH);	5,1
20	digitalWrite(10,HIGH);	5,2
21	digitalWrite(11,HIGH);	5,1
22	digitalWrite(12,HIGH);	5,2
23	digitalWrite(13,HIGH);	5,1

Tabel 4.3 Pengujian Pin Arduino Kondisi Logika 0

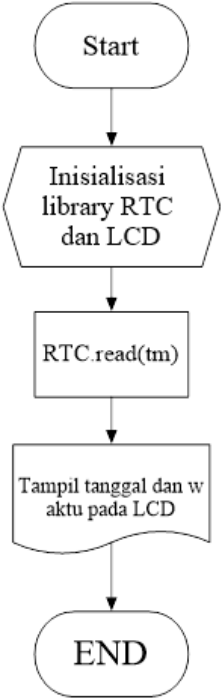
No	Pin dan Status	V Output
1	digitalWrite(A0,HIGH);	0.7 mV
2	digitalWrite(A1,HIGH);	0.7 mV
3	digitalWrite(A2,HIGH);	0.7 mV
4	digitalWrite(A3,HIGH);	0.5 mV
5	digitalWrite(A4,HIGH);	0.7 mV
6	digitalWrite(A5,HIGH);	0.21 mV
7	digitalWrite(A6,HIGH);	0.52 mV
9	digitalWrite(A7,HIGH);	0.21 mV
10	digitalWrite(0,HIGH);	0.46 mV
11	digitalWrite(1,HIGH);	0.3 mV
12	digitalWrite(2,HIGH);	0.3 mV
13	digitalWrite(3,HIGH);	0.23 mV
14	digitalWrite(4,HIGH);	0.45 mV
15	digitalWrite(5,HIGH);	0.23 mV
16	digitalWrite(6,HIGH);	0.23 mV
17	digitalWrite(7,HIGH);	0.21 mV
18	digitalWrite(8,HIGH);	0.4 mV
19	digitalWrite(9,HIGH);	0.3 mV
20	digitalWrite(10,HIGH);	0.52 mV
21	digitalWrite(11,HIGH);	0.4 mv
22	digitalWrite(12,HIGH);	0.3 mV
23	digitalWrite(13,HIGH);	0.4 mV

Pada Tabel 4.2 dan Tabel 4.3 dapat disimpulkan bahwa tegangan output pada Arduino mega dalam keadaan *high* dan *low* sudah sesuai dengan output yang diharapkan dan dapat digunakan. Pengukuran ini dibantu oleh avo meter .

4.3 Pengujian *Real Time Clock* (RTC)

Pengujian pada RTC dilakukan dengan cara menampilkan secara langsung waktu yang terbaca pada LCD 16x2 dengan membandingkan data waktu pada GMT di laptop.Untuk dapat menampilkan waktu dan

tanggal pada LCD 16x2 dilakukan terlebih dahulu memprogram RTC dan LCD menggunakan arduino mega. Pemograman menampilkan waktu dan tanggal menggunakan flowchart seperti pada gambar 4.4. Dengan demikian, diperoleh data pada Tabel 4.4 dan Gambar 4.4 sampai dengan Gambar 4.7 berupa tampilan waktu LCD16x2 dengan perbandingan waktu di laptop yaitu sebagai berikut.



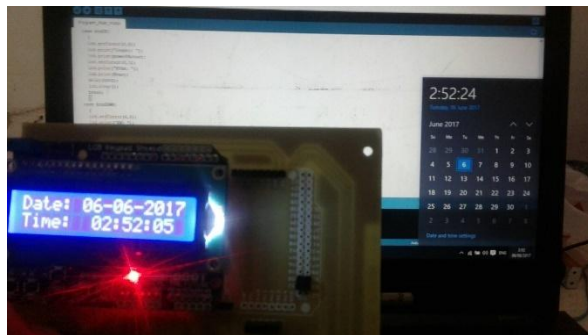
Gambar 4.4 Flowchart Pemograman Untuk Pengujian RTC.

Tabel 4.4 Pengujian Data Waktu pada RTC

No	Waktu pada RTC	Waktu pada Smartphone	Selisih Waktu
1	02:40:40	02:40:58	17 detik
2	02:52:05	02:52:05	17 detik
3	02:58:01	02:58:18	17 detik



Gambar 4.5 Pengujian Data Waktu pada RTC Data 1



Gambar 4.6 Pengujian Data Waktu pada RTC Data 2



Gambar 4.7 Pengujian Data Waktu pada RTC Data 3

Dengan mengacu data pada gambar dan tabel tersebut, didapatkan bahwa selisih antara jam RTC dan jam sesungguhnya

(Laptop) adalah konstan yaitu 17 detik. Sehingga dapat disimpulkan bahwa RTC dapat digunakan sebagai acuan.

4.4 Pengujian LCD (*Liquid Crystal Display*)

LCD dibutuhkan untuk menampilkan informasi data yang telah diolah. LCD yang digunakan ialah LCD 16x2. Dengan kata lain ada 16 kolom dan 2 baris yang tersedia untuk menampilkan suatu data yang terbaca dari sensor ataupun modul dengan dibantu arduino. Berikut hasil pengujian LCD pada Gambar 4.8 dan 4.9



Gambar 4.8 Pengujian Tampilan LCD dengan Satu Baris

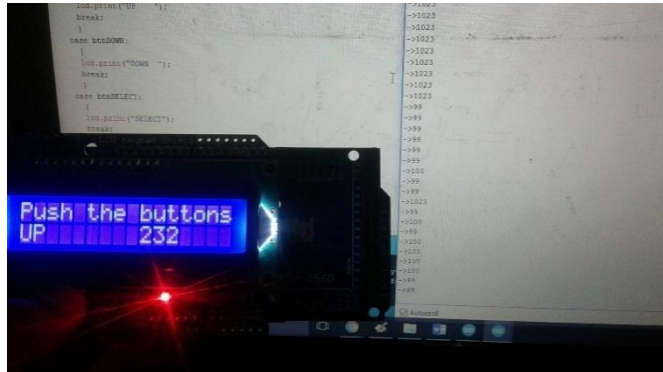


Gambar 4.9 Pengujian Tampilan LCD dengan Dua Baris

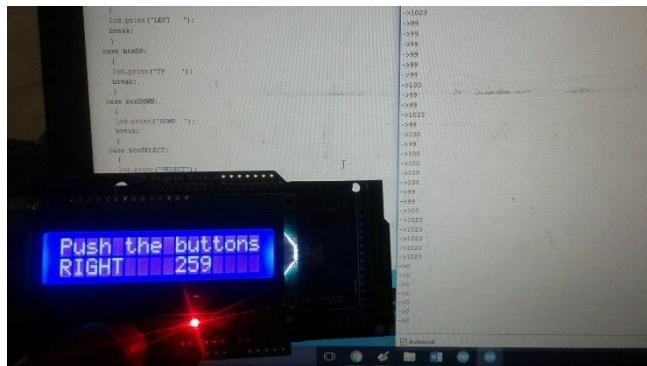
Dengan hasil yang sesuai maka LCD yang uji dapat digunakan untuk menampilkan data dari LCD.

4.5 Pengujian Keypad Module

Pengujian keypad matrix ini memastikan bahwa semua tombol keypad berfungsi dengan baik dan sesuai dengan perubahan tampilan pada LCD. Tampilan awal LCD sebelum keypad ditekan terlihat pada Gambar 4.10 dan 4.11 sebagai berikut.



Gambar 4.10 Pengujian Tampilan Push Button Kondisi 1



Gambar 4.11 Pengujian Tampilan Push Button Kondisi 2

Diperlihatkan bahwa saat menekan keypad tombol yang telah tersedia, maka tampilan pada LCD akan berubah sesuai inisialisasi tombol. Dan juga nominal ADC pada serial monitor yang berbeda.

4.6Pengujian Sensor Arus

Pengambilan data akurasi pembacaan sensor arus. Pengambilan data ini dilakukan dengan membandingkan data dari sensor arus yang terbaca di LCD dan besarnya arus yang terukur pada *clamp* meter seperti pada Gambar 4.12 sebagai berikut.

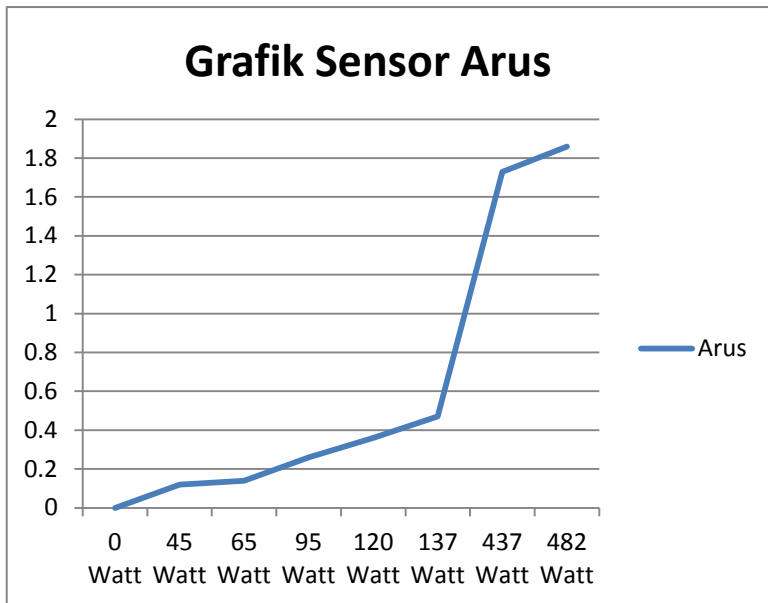


Gambar 4.12 Pengujian Perbandingan Hasil Sensor Arus dan Alat Ukur Arus Clamper

Ditunjukkan oleh Gambar4.12 proses pengambilan data arus dengan cara membandingkan pembacaan alat ukur clamper denganalat ukur sensor arus CT. Berikut hasil dari pembacaan arus pada Tabel 4.5:

Tabel 4.5 Data Pengambilan Data Arus

No	Clampmeter (Ampere)	Tampilan LCD (Ampere)	Error
1	0	0	0.0%
2	0.14	0.12	0.2%
3	0.14	0.14	0.0%
4	0.21	0.22	3.1%
5	0.35	0.36	2.8%
6	0.46	0.47	2.1%
7	1.72	1.73	0.6%
8	1.89	1.86	1.6%

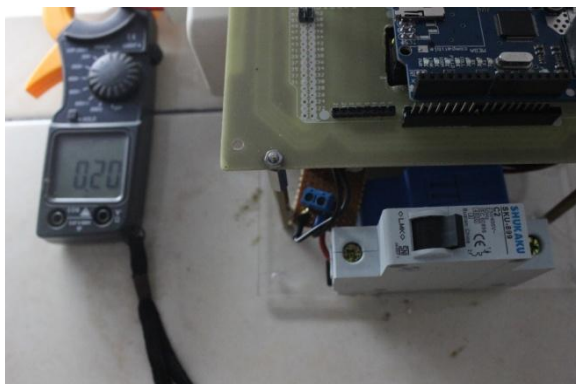


Gambar 4.13 Grafik Pembacaan Sensor Arus

Data yang telah diambil dengan pemakaian beban seperti pada Gambar 4.13 menunjukkan data error antara perbandingan dengan alat ukur arus *clammer* dengan data yang dibaca oleh sensor arus CT yang ditampilkan di LCD menunjukkan nilai error kurang lebih 3%. Dengan begitu alat ukur bisa digunakan untuk mengukur linearisasi arus.

4.7 Pengujian MCB

Pengujian MCB bertujuan untuk mengetahui apabila arus yang terukur lebih dari 2 Ampere maka MCB akan *trip* karena menggunakan MCB dengan batas arus 2 A. Beban yang digunakan dalam pengujian ini yaitu beban lampu 42 Watt, Setrika 300 Watt, Kipas angin 45 Watt sebanyak 3 buah. MCB 2 A terhubung dengan beban secara paralel. Pengujian MCB terdapat pada Gambar 4.14 sebagai berikut.



Gambar 4.14 Pengujian MCB

Hasil dari pengujian MCB tersebut, pada arus dibawah 2,2 Ampere MCB masih terhubung. Ketika pembacaan arus oleh *clamp* meter sebesar 2,2 Ampere maka MCB *trip*. Pengujian besarnya arus dengan clamp meter terdapat pada gambar 4.13 dan diperoleh hasil seperti pada Tabel 4.6 sebagai berikut.

Tabel 4.6 Hasil Pengambilan Data Trip MCB 2 A

No	Arus	Kondisi
1	0	Terhubung
2	0,1	Terhubung
3	0,20	Terhubung
4	0,47	Terhubung
5	1,80	Terhubung
6	1,97	Terhubung
7	2,02	Terhubung
8	2,16	Trip

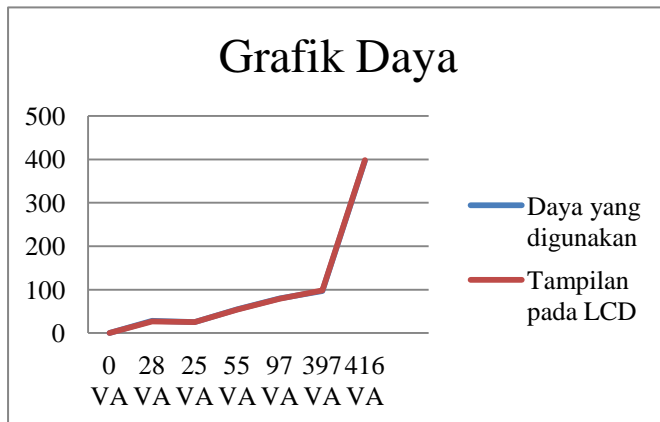
4.8 Pengujian Perhitungan VA

Pada alat KVAR Meter ini selain untuk mengukur suatu daya reaktif, Daya semu atau daya total (S), ataupun juga dikenal dalam Bahasa Inggris Apparent Power, adalah hasil perkalian antara tegangan efektif (root-mean-square) dengan arus efektif (root-mean-square). Berikut pengambilan data dari perhitungan beban dengan membandingkan jumlah beban yang digunakan dan nominal yang ditunjukkan pada LCD

Tabel 4.7 Tabel Pengukuran Daya Kompleks

No	Jumlah Beban (VA)	Tampilan Apparent Power LCD (VA)	Error
1	0	0	0.0%
2	28	26.9	4.1%
3	25	25.41	1.6%
4	55	53.72	2.4%
5	97	97.95	1.0%
6	397	398.32	0.3%
7	416	414.21	0.4%

Pada Tabel 4.7 dapat dilihat menggunakan beban dari 28 VA hingga 416 VA. Jenis beban yang digunakan yaitu 3 lampu bohlam, 1 Setrika, dan 2 kipas angin Dan didapat error terbesar yaitu 4%. Proses pengambilan data ini dilakukan sebanyak 7 sampling beban yang berbeda. Berikut tampilan grafik daya yang ditunjukkan pada Gambar 4.15



Gambar 4.15 Perbandingan Pengukuran Daya

Untuk proses pengukuran beban yang terdapat pada Gambar 4.16 dapat dilihat pada Tabel 4.7 bahwa nilai nominal error hanya 2%. Dengan begitu pengukuran nilai data daya dapat digunakan untuk pengukuran daya dengan baik. Pada kurva beban pula ditunjukkan letak error yang sangat sedikit tidak mempengaruhi dari kurva pada daya yang digunakan. Dengan begitu bentuk data yang telah diambil sudah baik.



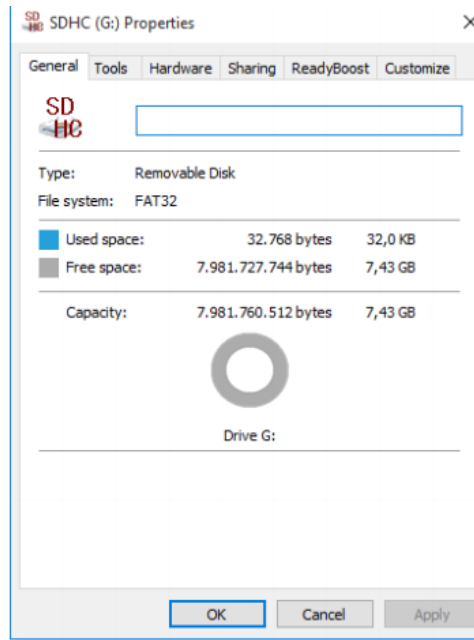
Gambar 4.16 Pengambilan Data Membandingkan Tampilan pada LCD dengan Watt yang Digunakan

4.9 Pengujian Memori SD Card (Data Logger)

Pada pengujian SD Card dilakukan untuk mengetahui kapasitas yang dapat ditampung oleh SD Card. Memori yang digunakan adalah MicroSD Card VGen dengan kapasitas penyimpanan 8 Gb. Pengujian dilakukan dengan pembacaan kapasitas kartu pada komputer dalam kondisi kosong. Pengujian dilakukan untuk memastikan SD Card memiliki ruang penyimpanan yang benar-benar kosong, yang nantinya akan diisi data arus dan tegangan. Berikut adalah data SD Card yang telah diambil dan terbagi menjadi beberapa bagian yang seperti pada Tabel 4.8

Tabel 4.8 Pengujian SD Card

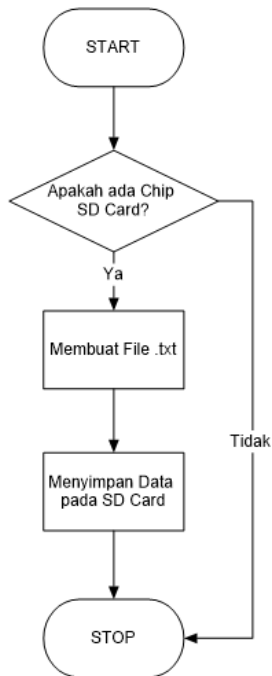
Nama Ruang	Ukuran (bytes)	Ukuran
Used Space	32.768	32.0 KB
Free Space	7.981.727.744	7.43 GB
Capacity	7.981.760.512	7.43 GB



Gambar 4.17 Hasil Pengujian SD Card KVAR Meter

Dengan demikian maka kondisi SD Card dapat digunakan pada Tugas Akhir ini untuk menyimpan nilai arus, dan tegangan pada saat kondisi normal dapat dilihat pada Gambar 4.17. Selain dengan memeriksa memori kartu, dilakukan juga pengujian penyimpanan data arus dan tegangan ke dalam *filetext* (.txt). Gambar 4.18 adalah gambar *flowchart* untuk penyimpanan data pada *file text* (.txt). Agar file dapat dijadikan sebuah data yang dapat diproses, maka file .txt diubah kedalam file .xlsx (Microsoft Excel). Langkah langkah untuk mengubah data .txt menjadi .xlsx adalah sebagai berikut :

1. Buka Microsoft Excel
2. Buka menu Data
3. Pada Submenu “*Get External Data*”, pilih “*From Text*”
4. Selanjutnya pilih file .txt penyimpanan data dari Arduino
5. Akan muncul jendela baru “*Text Import Wizard*”, pilih *Delimited* lalu *Next*
6. Lalu pilih *Delimited* (pemisah data) yang digunakan pada file .txt, lalu pilih *Next*.
7. Pada data *format* pilih *General*. Tekan *Finish*
8. Pada jendela Import Data centang “*Add this data to the Data Model*” lalu klik OK.



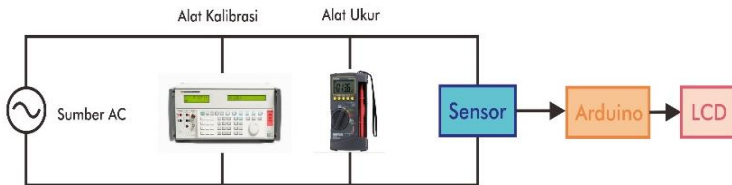
Gambar 4.18 Flowchart Penyimpanan Data .txt pada SD Card

Pengambilan data penyimpanan data dilakukan pada SD Card dengan data KVARH meter.

4.10 Pengujian Perhitungan Sensor Tegangan

Pada pengujian sensor tegangan ini kami melakukan kalibrasi sensor tegangan untuk mendapatkan tingkat akurasi pembacaan arus

sebaik mungkin. Pengujian dilakukan kepada trafo CT 350 mA yang di gunakan yang dengan menggunakan alat kalibrasi FLUKE 5500A.seperti pada Gambar 4.19



Gambar 4.19 Skema Pengujian Sensor Tegangan

Tabel 4.9 Pembacaan Sensor Tegangan

No	FLUKE 5500A (Volt)	Sensor Tegangan (Volt)	Error
1	160	161,19	0,74
2	170	170,99	0,58
3	180	181,12	0,62
4	190	191,04	0,55
5	200	201,25	0,63
6	210	211,25	0,60
7	220	221,28	0,58

Dari data pada Tabel 4.9 yang telah diperoleh dari pengujian sensor tegangan, dapat diketahui bahwa nilai *error* sensor tegangan sebesar 0,55% Dengan begitu, sensor tegangan ini dapat digunakan dengan baik, karena tidak memiliki nilai error yang besar.

4.11 Pengujian Perhitungan Cosphi

Diketahui nilai cosphi dibutuhkan untuk mengetahui daya resistif pada suatu beban. Cosphi ini sangat diperhatikan oleh pihak PLN akibat rugi-rugi daya. Konsumen harus mengatur nilai cosphi agar tidak terkena denda oleh PLN. Berpengaruhnya bagus buruknya nominal cosphi dipegaruhi oleh beban induktif yang digunakan. Nominal minimal nominal cosphi yang harus ditaati yaitu tidak boleh kurang dari 0,85. Dengan begituu pengujian cosphi sangat dibutuhkan. Berikut nilai pengujian cosphi

Tabel 4.10 Hasil Pembacaan Cosphi Meter dengan LCD

No	Beban	Jenis	Tampilan Cosphi LCD	Cosphi Meter	Error
1	300 Watt	Rice cooker	0,97	1,00	3%
2	300 Watt	Setrika	0,98	1,00	2%
3	50 Watt	Lampu Dop	0,98	0,96	2%

Pada Tabel 4.10 dapat dilihat perbandingan pengukuran alat ukur cosphi meter dan hasil yang ditunjukan pada LCD tidak beda jauh. Error yang didapat kurang dari 2%. Maka dari itu hasil ini bisa dijadikan acuan pengukuran cosphi. Proses pengambilan data cosphi dapat dilihat pada Gambar 4.20.



Gambar 4.20 Hasil Pembacaan Cosphi Meter

4.12 Pengujian Keseluruhan KVARH Meter

Pengujian keseluruhan KVARH meter ini bertujuan untuk memeriksa apakah system yang dibuat sudah diinginkan. Pengujian dari nominal cosphi, nilai VAR, Watt, Notifikasi kepada pelanggan, hingga rekapitulasi perbulannya. Berikut tabel dan gambar hasil pengujian keseluruhan:

Tabel 4.11 Tabel Pembacaan Beban yang Digunakan

NO	BEBAN	JENIS	COSPHI	Watt	VAR
1	300 Watt	<i>Rice cooker</i>	0,97	291	73.7
2	600 Watt	<i>Rice cooker + Setrika</i>	0,98	588	120. 2
4	300 Watt	Setrika	0,98	294	60.1
5	50 Watt	Lampu Dop	0,98	49	10.0
6	625 Watt	<i>Rice Cooker + Setrika+ Lampu DOP</i>	0,99	619	88.5
7	310 Watt	Lampu DOP + <i>Rice Cooker</i>	0,98	304	62.1
8	85 Watt	Lampu Dop+Lampu TL	0,88	75	42.1
9	35 Watt	Lampu Dop+Lampu TL	0,67	23	29.3
10	10 Watt	Lampu TL	0,42	4	11.4

Dapat dilihat dari hasil tabel pengukuran cosphi beban, bahwa beban yang digunakan ada yang mempunyai nominal cosphi yang kurang baik yaitu lampu TL. Nominal cosphi itu yang nantinya mempengaruhi sistem pemberitahuan dan rekapitulasi penggunaan beban induktif. Dalam perhitungan daya reaktif dan daya aktif belum dikalikan dengan perhitungan kalkulasi dalam satuan kilo dan perhitungan perjamnya



Gambar 4.21 Proses Pengambilan Cosphi pada Jenis Beban

Beban yang terukur akan terhubung langsung dengan IOT. IOT akan mengolah data, data yang akan diolah yaitu rekapitulasi dan batas notifikasi yang bertujuan untuk pelanggan agar bisa memantau penggunaan beban .

Dashboard

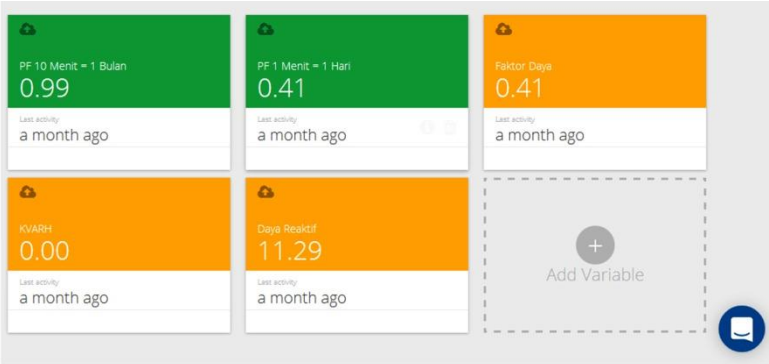
PF 1 Menit = 1 Hari

Date	PF 1 Menit = 1 Hari
June 16 2017 at 14:53:00	0.40638
June 16 2017 at 14:52:00	0.15272499999999997
June 16 2017 at 14:51:00	0.832757142857143
June 16 2017 at 14:50:00	0.98711

Faktor Daya

Date	Faktor Daya
------	-------------

Gambar 4.22 Interface pada UBIDOTS

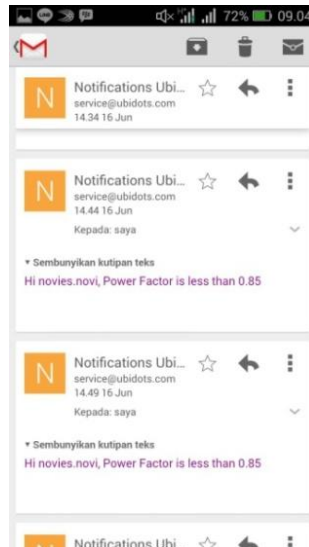


Gambar 4.23 Proses Pengolahan Data padaUBIDOTS

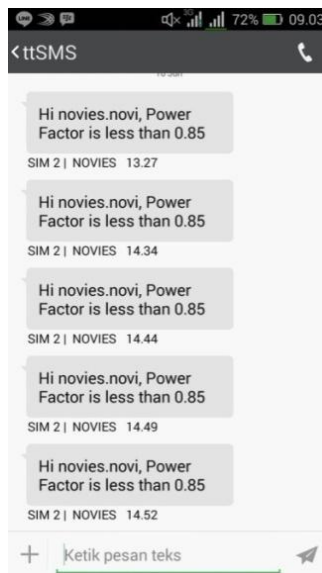
PF 10 Menit = 1 Bulan			
Date	PF 10 Menit = 1 Bulan		
June 16 2017 at 14:50:00	0.4967405882352942		
June 16 2017 at 14:40:00	0.8970113953488368		
June 16 2017 at 14:30:00	0.7015324999999998		
June 16 2017 at 14:20:00	0.7821331746031746		
June 16 2017 at 14:10:00	0.09199744680851062		
June 16 2017 at 13:50:00	0.265735		
June 16 2017 at 13:30:00	0.3771320930232559		
June 16 2017 at 13:20:00	0.38397181818181814		

Gambar 4.24 Tampilan Rekapitulasi Sudut Cosphi dalam Satu Bulan

Pada rekapitulasi ini menggunakan penyederhanaan perhitungan waktu yaitu untuk waktu pengambilan data selama 10 menit diibaratkan waktu selama sebulan. Dalam pengimplementasian dalam kehidupan sehari-hari, bertujuan agar pelanggan mengetahui rekapan cosphi selama penggunaan dalam durasi satu bulan. Diharapkan pelanggan dapat mengatur penggunaan beban setelah mengetahui hasil rekapitulasi beban yang digunakan terutama beban induktif



Gambar 4.25 Tampilan Notifikasi via Email Pelanggan



Gambar 4.26 Tampilan Notifikasi Cosphi via SMS Kepada Pelanggan

4.12 Analisa Relevansi

Alat merupakan *prototype* awal KVARH meter dengan media yang terintegreasi oleh IoT. Jika digunakan pada kehidupan sehari-hari pada dunia industri, perlu banyak yang disesuaikan kembali. Pada *prototype* ini hanya menggunakan permodelan satu fasa, dengan sumber tegangan 220 Volt dan menggunakan batasan arus dengan MCB sebesar 2A. Pada kehidupan sehari-hari bahwa pelanggan industri menggunakan sumber 3 fasa. Dan permodelan perhitungan KVARH dengan satuan waktu berupa jam lebih dipersingkat menjadi menit agar dapat melihat hasil lebih cepat tanpa mengurangi keakuratan perhitungan. Kemudian untuk system rekapitulasinya pun menggunakan penyederhanaan yang seharusnya perhitungan dalam waktu dalam jam menjadi menit.

-----Halaman ini sengaja dikosongkan-----

BAB V

PENUTUP

Setelah melakukan perencanaan, pembuatan, dan pengujian sistem *KVARH Meter*, maka ditarik kesimpulan dan saran dari kegiatan yang telah dilakukan.

5.1 Kesimpulan

Dari Tugas Akhir yang telah dilakukan, dapat diambil beberapa kesimpulan sebagai berikut.

1. *KVAR Meter* merupakan VAR meter yang memiliki fitur tambahan yaitu dapat melakukan *monitoring* arus, tegangan, *cos phi*, nominal Watt, nominal VA.
2. Alat *Kvar Meter* ini dapat mengakumulasi pengeluaran beban yang terpasang pada pelanggan, selain itu bentuk akumulasi *cosphi*nya yang digunakan oleh pelanggan
3. Untuk pengukuran *cosphi* menggunakan *sampling* gelombang dan membandingkan bentuk gelombang pada arus dan tegangannya.
4. *Sampling* gelombang digunakan untuk melihat *leading* dan *legging* gelombang yang tercatat dari *sample* yang dikirim dan dibagi oleh jumlah *sampling* yang dikirim.
5. Perhitungan *cosphi* yang diukur oleh *prototype KVARH meter* ini kurang dari 4%
6. Perhitungan daya imajiner yang diukur oleh *prototype KVARH meter* ini kurang dari 5%
7. *Prototype* ini digunakan pada pembebanan satu fasa dengan sumber 220 V

5.2 Saran

Dengan memperhatikan beberapa kelemahan dari proyek Tugas Akhir ini, maka diberikan beberapa saran yang sekiranya dapat dikembangkan pada masa yang akan datang demi kesempurnaan dari proyek tugas akhir ini. Adapun beberapa saran tersebut yaitu:

1. Dikarekanakan alat ini berupa perhitungan data disarankan pengambilan sampel data, linierisasi data sensor, dan pengujian alat secara keseluruhan lebih banyak dan berulang, untuk memastikan kemampuan alat serta keakuratan alat.
2. Dalam pengujian hendaknya menggunakan alat ukur pembanding digital atau yang lebih akurat agar hasil pengujian bisa mendekati nilai yang sebenarnya.

3. Perlunya ada pengembangan di metode perhitungan daya tegangan karena metode sampling gelombang perlunya bentuk gelombang yang bagus dan akurat.

DAFTAR PUSTAKA

- [1], **PENGESAHAN RUPTL PLN TAHUN 2016-2025**, MENTERI ENERGI DAN SUMBER DAYA MINERAL REPUBLIK INDONESIA, 2016.
- [2], **BUKU PANDUAN PUSDIKLAT PLN** , TRANSFER ENERGI LISTRIK, 2016
- [3], **IDE Software for Arduino**, Datasheet, 2015.
- [4] Andrianto, Heri; dan Aan Darmawan. 2016. **Arduino Belajar Cepat dan Pemrograman**. Bandung : INFORMATIKA.
- [5], **OPENENERGY MONITOR**, <https://openenergymonitor.org/>, 1 Mei 2017
- [6] *LPKEE ITB's students blog*, <https://konversi.wordpress.com>, 17 April 2017
- [7], **INFORCE UNINTERRUPTIBLE POWER SUPPLY 650 VA**, Manual Book, 2014.
- [8] DFROBOT, **ARDUINO LCD KEYPAD DFROBOT**, <https://www.dfrobot.com/>, 19 Mei 2017.
- [8] Nafita H. S., dan Rifki N., "*Smart Electricity Meter*", *Tugas Akhir*, Institut Teknologi Sepuluh Nopember, Surabaya, 2016.
- [9] DFROBOT, **ARDUINO LCD KEYPAD DFROBOT**, <https://www.dfrobot.com/>, 19 Mei 2017.
- [10] Ayu Dyah S., "Monitoring Beban KWH Meter 1 Phasa pada pelanggan rumah Tangga dalam Satu Wilayah", *Tugas Akhir*, Institut Teknologi Sepuluh Nopember, Surabaya, 2013.
- [11] Asrul S., "Rancang Bangun Alat untuk Perbaikan Faktor Daya pada Beban Dinamis 1 Fase dan Monitoring Daya dengan LCD Grafik", Politeknik Elektronika Negeri Surabaya, , Surabaya, (tanpa tahun).

-----Halaman ini sengaja dikosongkan-----

LAMPIRAN A

```
#include <Ethernet.h>
#include <SPI.h>
#include <UbidotsEthernet.h>
#define ID1 "596d8a17762542090697f5fa" //
#define ID2 "593e3ac37625420179688ab3"
#define ID3 "593e3a7e76254201796886dd"
#define TOKEN "ubsHI8i96Khm77sf5PrR9GHJ3RVYdv" //
#define cosh
#define atan
#include "EmonLib.h"
#include <LiquidCrystal.h>
#include <DS1307RTC.h>
#include <Wire.h>
#include <Time.h>
#include <DS1307RTC.h>
#include <math.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
// Include Emon Library
EnergyMonitor emon1;

// Enter a MAC address for your controller below.
// Newer Ethernet shields have a MAC address printed on a sticker
on the shield
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
// Set the static IP address to use if the DHCP fails to assign
IPAddress ip(192,168,1,9);

Ubidots client(TOKEN);
void setup()
{
  Serial.begin(9600);
  lcd.begin(16,2);
  emon1.voltage(1, 182.21, 1.7); // Voltage: input pin, calibration,
phase_shift
  emon1.current(3, 14.781); // Current: input pin, calibration.
  if (Ethernet.begin(mac) == 0) {
    Serial.println("Failed to configure Ethernet using DHCP");
```

```

    // try to configure using IP address instead of DHCP:
    Ethernet.begin(mac, ip);
  }
  // give the Ethernet shield a second to initialize:
  delay(1000);
}
void loop()
{
  emon1.calcVI(80,8000);          // Calculate all. No.of half
  wavelengths (crossings), time-out
  emon1.serialprint();           // Print out all variables (realpower,
  apparent power, Vrms, Irms, power factor)

  float realPower    = emon1.realPower;    //extract Real Power
  into variable
  float apparentPower = emon1.apparentPower;    //extract
  Apparent Power into variable
  float powerFactor   = emon1.powerFactor;    //extract Power
  Factor into Variable
  float supplyVoltage = emon1.Vrms;           //extract Vrms into
  Variable
  float Irms          = emon1.Irms;           //extract Irms into
  Variable
  float Var           = sqrt((apparentPower*apparentPower)-
  (realPower*realPower));
  float KWH = KWH + (apparentPower * (2.05/60/1000));
  //Calculate kilowatt hours used
  float KVARH = KVARH + ( Var * (2.05/60/1000));
  //float COSPHIR = double cos( atan * KVARH/KWH);
  //float cosphihari = double cos (double__(double atan
  (double__KWH/KVARH)));
  float kwhkvarh = KWH/KVARH;
  float tangen = atan (kwhkvarh);
  float cosphihari = cos (tangen);

  Serial.print("Arus: ");
  Serial.println(Irms);
  Serial.print("Tegangan: ");
  Serial.println(supplyVoltage);
  Serial.print("daya nyata: ");
  Serial.println(realPower);

```

```

Serial.print("Daya semu: ");
Serial.println(apparentPower);
Serial.print("power factor: ");
Serial.println(powerFactor);
Serial.print("Reaktif ");
Serial.println(Var);
Serial.print("KWH :");
Serial.println(KWH);
Serial.print("KVARH: ");
Serial.println(KVARH);
Serial.print("Cosphihari: ");
Serial.println(cosphihari);
lcd.setCursor(0,0);
lcd.print("KVARH: ");
lcd.print(KVARH);
lcd.setCursor(0,1);
lcd.print("cosphi: ");
lcd.print(powerFactor);
delay(8000);
client.add(ID1, powerFactor);
client.add(ID2, KVARH);
client.add(ID3, Var);
client.sendAll();
}

```

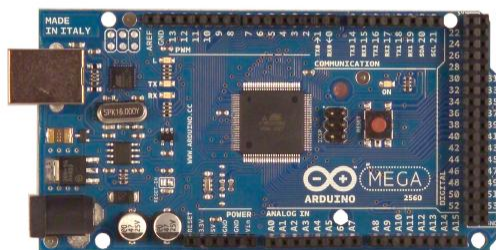
-----Halaman ini sengaja dikosongkan-----

LAMPIRAN B

B.1 DATASHEET ARDUINO MEGA



Arduino Mega 2560 Datasheet



Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2).** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 0 to 13.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** These pins support SPI communication using the [SPI library](#). The SPI pins are also broken out on the ICSP header, which is physically compatible with the Uno, Duemilanove and Diecimila.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH

value, the LED is on, when the pin is LOW, it's off.

- **I²C: 20 (SDA) and 21 (SCL).** Support I²C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I²C pins on the Duemilanove or Decimila.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and `analogReference()` function.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with `analogReference()`.
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Mega2560's digital pins.

The ATmega2560 also supports I²C (TWI) and SPI communication. The Arduino software includes a `Wire` library to simplify use of the I²C bus; see the [documentation on the Wiring website](#) for details. For SPI communication, use the [SPI library](#).

Programming

The Arduino Mega can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Arduino Mega comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It

communicates using the original STK500 protocol ([reference](#), [C header files](#)).

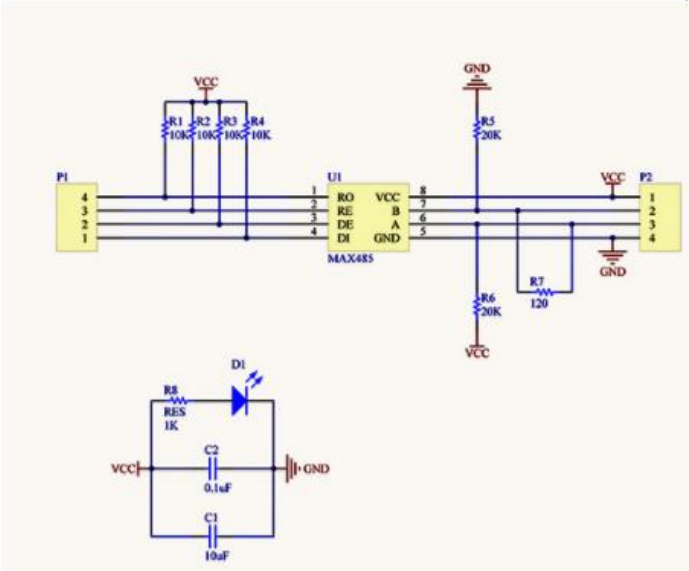
You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

B.2DATASHEET RS485

How-To Information Link HERE:

See Example RS485 Network Diagram below.

SCHEMATIC OF THIS MODULE:



Selection Table

PART NUMBER	HALF/FULL DUPLEX	DATA RATE (Mbps)	SLEW-RATE LIMITED	LOW-POWER SHUTDOWN	RECEIVER/ DRIVER ENABLE	QUIESCENT CURRENT (μA)	NUMBER OF RECEIVERS ON BUS	PIN COUNT
MAX481	Half	2.5	No	Yes	Yes	300	32	8
MAX483	Half	0.25	Yes	Yes	Yes	120	32	8
MAX485	Half	2.5	No	No	Yes	300	32	8
MAX487	Half	0.25	Yes	Yes	Yes	120	128	8
MAX488	Full	0.25	Yes	No	No	120	32	8
MAX489	Full	0.25	Yes	No	Yes	120	32	14
MAX490	Full	2.5	No	No	No	300	32	8
MAX491	Full	2.5	No	No	Yes	300	32	14
MAX1487	Half	2.5	No	No	Yes	230	128	8

For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim Integrated's website at www.maximintegrated.com.

19-0122; Rev 10; 9/14

B.3 DATASHEET RTC DS1307

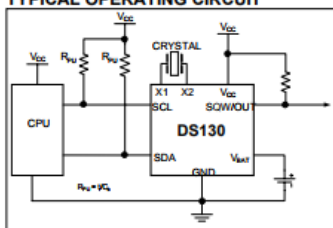


DS1307 64 x 8, Serial, I²C Real-Time Clock

GENERAL DESCRIPTION

The DS1307 serial real-time clock (RTC) is a low-power, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially through an I²C, bidirectional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power-sense circuit that detects power failures and automatically switches to the backup supply. Timekeeping operation continues while the part operates from the backup supply.

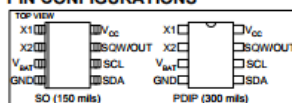
TYPICAL OPERATING CIRCUIT



FEATURES

- Real-Time Clock (RTC) Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the week, and Year with Leap-Year Compensation Valid Up to 2100
- 56-Byte, Battery-Backed, General-Purpose RAM with Unlimited Writes
- I²C Serial Interface
- Programmable Square-Wave Output Signal
- Automatic Power-Fail Detect and Switch Circuitry
- Consumes Less than 500nA in Battery-Backup Mode with Oscillator Running
- Optional Industrial Temperature Range: -40°C to +85°C
- Available in 8-Pin Plastic DIP or SO
- Underwriters Laboratories (UL) Recognized

PIN CONFIGURATIONS



ORDERING INFORMATION

PART	TEMP RANGE	VOLTAGE (V)	PIN-PACKAGE	TOP MARK*
DS1307+	0°C to +70°C	5.0	8 PDIP (300 mils)	DS1307
DS1307N+	-40°C to +85°C	5.0	8 PDIP (300 mils)	DS1307N
DS1307Z+	0°C to +70°C	5.0	8 SO (150 mils)	DS1307
DS1307ZN+	-40°C to +85°C	5.0	8 SO (150 mils)	DS1307N
DS1307Z+T&R	0°C to +70°C	5.0	8 SO (150 mils) Tape and Reel	DS1307
DS1307ZN+T&R	-40°C to +85°C	5.0	8 SO (150 mils) Tape and Reel	DS1307N

*Denotes a lead-free/RoHS-compliant package.

*A "+" anywhere on the top mark indicates a lead-free package. An "N" anywhere on the top mark indicates an industrial temperature range device.

ABSOLUTE MAXIMUM RATINGS

Voltage Range on Any Pin Relative to Ground	-0.5V to +7.0V
Operating Temperature Range (Noncondensing)	
Commercial	0°C to +70°C
Industrial	-40°C to +85°C
Storage Temperature Range	-55°C to +125°C
Soldering Temperature (DIP, leads)	+260°C for 10 seconds
Soldering Temperature (surface mount)	Refer to the JPC/JEDEC J-STD-020 Specification.

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to the absolute maximum rating conditions for extended periods may affect device reliability.

RECOMMENDED DC OPERATING CONDITIONS

(T_A = 0°C to +70°C, T_A = -40°C to +85°C.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Supply Voltage	V _{CC}		4.5	5.0	5.5	V
Logic 1 Input	V _{HI}		2.2		V _{CC} + 0.3	V
Logic 0 Input	V _{LI}		-0.3		+0.8	V
V _{BAT} Battery Voltage	V _{BAT}		2.0	3	3.5	V

DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 4.5V to 5.5V; T_A = 0°C to +70°C, T_A = -40°C to +85°C.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Input Leakage (SCL)	I _I		-1		1	μA
I/O Leakage (SDA, SQW/OUT)	I _{LO}		-1		1	μA
Logic 0 Output (I _{OL} = 5mA)	V _{OL}				0.4	V
Active Supply Current (f _{SCL} = 100kHz)	I _{CCA}				1.5	mA
Standby Current	I _{CCS}	(Note 3)			200	μA
V _{BAT} Leakage Current	I _{BATLKG}			5	50	nA
Power-Fail Voltage (V _{BAT} = 3.0V)	V _{PF}		1.216 x V _{BAT}	1.25 x V _{BAT}	1.284 x V _{BAT}	V

DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 0V, V_{BAT} = 3.0V; T_A = 0°C to +70°C, T_A = -40°C to +85°C.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
V _{BAT} Current (OSC ON); SQW/OUT OFF	I _{BAT1}			300	500	nA
V _{BAT} Current (OSC ON); SQW/OUT ON (32kHz)	I _{BAT2}			480	800	nA
V _{BAT} Data-Retention Current (Oscillator Off)	I _{BATDR}			10	100	nA

WARNING: Negative undershoots below -0.3V while the part is in battery-backed mode may cause loss of data.

AC ELECTRICAL CHARACTERISTICS(V_{CC} = 4.5V to 5.5V; T_A = 0°C to +70°C, T_A = -40°C to +85°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SCL Clock Frequency	f _{SCL}		0		100	kHz
Bus Free Time Between a STOP and START Condition	t _{BUF}		4.7			μs
Hold Time (Repeated) START Condition	t _{HD,STA}	(Note 4)	4.0			μs
LOW Period of SCL Clock	t _{LOW}		4.7			μs
HIGH Period of SCL Clock	t _{HIGH}		4.0			μs
Setup Time for a Repeated START Condition	t _{SUSTA}		4.7			μs
Data Hold Time	t _{HDDAT}		0			μs
Data Setup Time	t _{SUDAT}	(Notes 5, 6)	250			ns
Rise Time of Both SDA and SCL Signals	t _R				1000	ns
Fall Time of Both SDA and SCL Signals	t _F				300	ns
Setup Time for STOP Condition	t _{SUSTO}		4.7			μs

CAPACITANCE(T_A = +25°C)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Pin Capacitance (SDA, SCL)	C _{IO}				10	pF
Capacitance Load for Each Bus Line	C _B	(Note 7)			400	pF

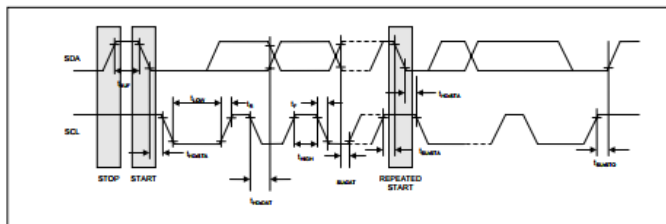
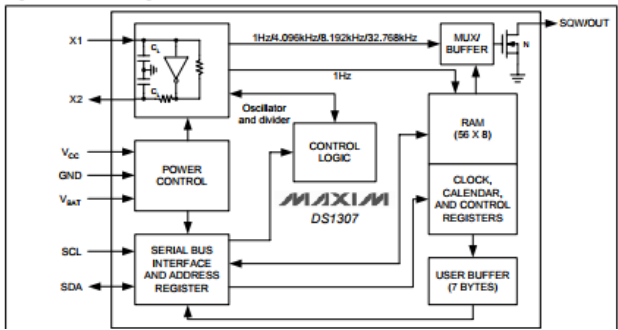

Note 1: All voltages are referenced to ground.**Note 2:** Limits at -40°C are guaranteed by design and are not production tested.**Note 3:** I_{CCS} specified with V_{CC} = 5.0V and SDA, SCL = 5.0V.**Note 4:** After this period, the first clock pulse is generated.**Note 5:** A device must internally provide a hold time of at least 300ns for the SDA signal (referred to the V_{IHQ} of the SCL signal) to bridge the undefined region of the falling edge of SCL.**Note 6:** The maximum t_{SUDAT} only has to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.**Note 7:** C_B—total capacitance of one bus line in pF.**TIMING DIAGRAM**

Figure 1. Block Diagram




B.4 DATASHEET YHDC SCT 13-010

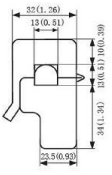


Product Specification

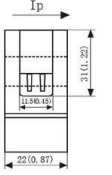
Date:2015-8-7

Product Name	Current transformer	Model	SCT013-010
Characteristics:Opening size 13mm*13mm,1m leading wire, standard Φ3.5 three-core plug output. Have two kinds of output type: Current output type and voltage output type. Purpose: Used for current measurement, monitor and protection for AC motor,lighting equipment, air compressor etc			
Technical Data			
I_{PN}	Rated input	0-10A	
I_{PM}	Max. detection input		
I_{OUT}	Rated output	0-1V	
X	Accuracy	±1%	
ϵ_L	Linearity	≤0.2%	
N	Turns ratio	1:1800	
Φ	Phase shift		
R_L	Max.Sampling resistance		
V_{PN}	Work voltage	660V	
f	Work frequency	50-1KHz	
T_A	Operating temperature	-25..+70℃	
T_S	Storage temperature	-40..+85℃	
Vd	Dielectric strength, 50 Hz, 1 min	3KV	
			
		Fire resistance	UL94-V0
		Material of core	Ferrite
		Mounting type	Suspension
		Weight	55g

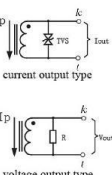
Dimension (mm(in), 1 mm= 0.0394 inch)



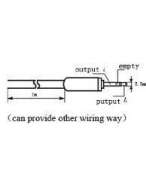
Front view



Side view

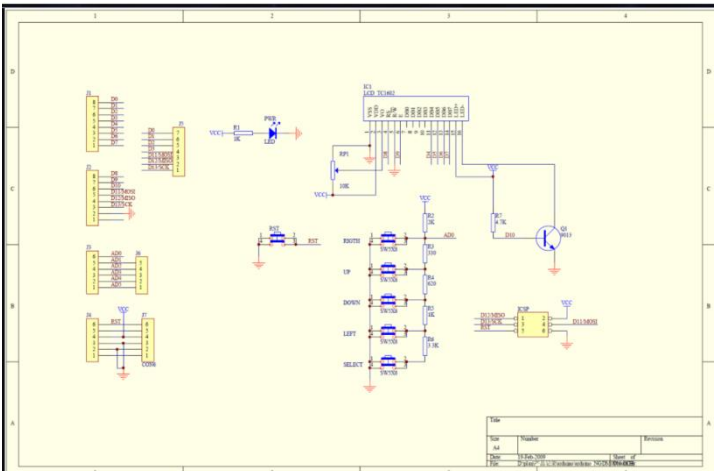


Schematic diagram

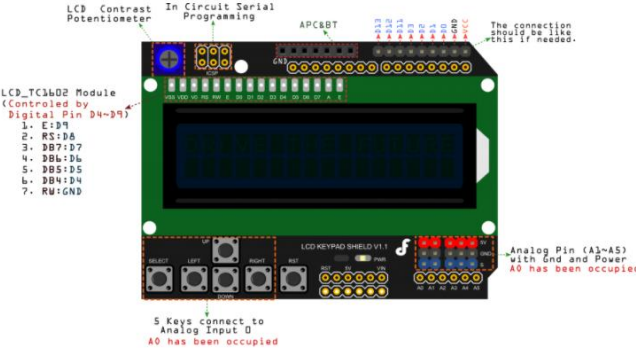


Standard three-core plugs schematic diagram

B.5 DATASHEET LCD Keypad Shield



Pinout

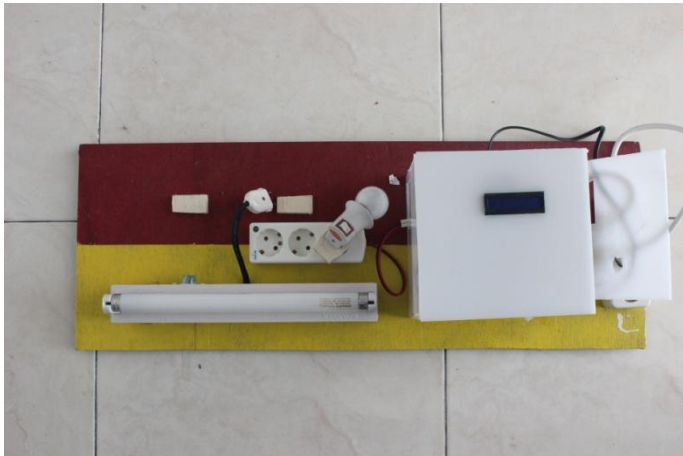


Instruction for D4 To D10 and Analog Pin 0		
Pin	Function	Instruction
Digital 4(D4)		
Digital 5(D5)	D4-D7 are used as DB4-DB7	Four high order bidirectional tristate data bus pins. Used for data transfer and receive between the MPU and the LCD.
Digital 6(D6)		
Digital 7(D7)		
Digital 8(D8)	RS	Choose Data or Signal Display
Digital 9(D9)	Enable	Starts data read/write
Digital 10(D10)	LCD Backlight Control	
Analog 0(A0)	Button select	Select, up, right, down and left

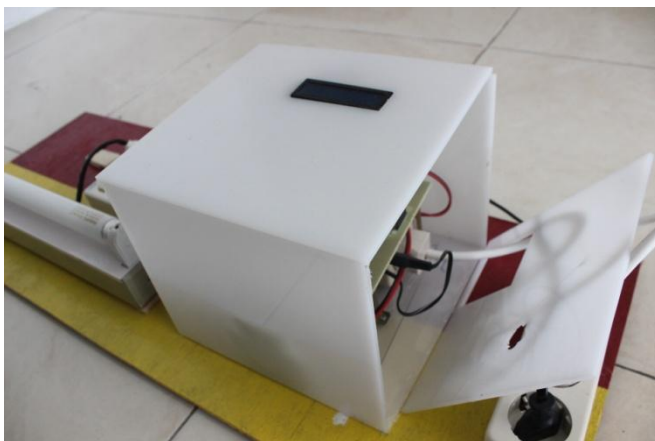
-----Halaman ini sengaja dikosongkan-----

LAMPIRAN C

C.1 ALAT PROTOTYPE KVARH METER TAMPAK ATAS



C.2 ALAT PROTOTYPE KVARH METER TAMPAK SAMPING



C.3 ALAT PROTOTYPE KVARH METER RANGKA DALAM



DAFTAR RIWAYAT HIDUP



Nama : Rahadian Yunanto D P
TTL : Surabaya, 01 April 1996
Jenis Kelamin : Laki - Laki
Agama : Islam
Alamat : Delta Sari Indah BL-33,
Waru, Sidoarjo
Telp/HP : 085856965797
E-mail : rahdian_ade@ymail.com

RIWAYAT PENDIDIKAN

1. 2002 – 2008 : SD Al-Falah Tropodo Sidoarjo
2. 2008 – 2011 : SMP Al-Falah Tropodo Sidoarjo
3. 2011 – 2014 : SMA Negeri 6 Surabaya
4. 2014 – 2017 : D3 Teknik Elektro Otomasi, Program Studi Teknik Listrik – Fakultas Vokasi Institut Teknologi Sepuluh Nopember (ITS)

PENGALAMAN KERJA

1. Kerja Praktek di PT PLN (Persero) Rayon Rungkut Area Surabaya Selatan Jawa Timur

-----Halaman ini sengaja dikosongkan-----